INTELLICHAIR: A NON-INTRUSIVE SITTING POSTURE AND SITTING ACTIVITY RECOGNITION SYSTEM

A thesis submitted for the degree of
Doctor of Philosophy (PhD)

by

Teng Fu

School of Science, Engineering and Technology
Abertay University.

May 2015

Abstract

Current Ambient Intelligence and Intelligent Environment research focuses on the interpretation of a subject's behaviour at the activity level by logging the Activity of Daily Living (ADL) such as eating, cooking, etc. In general, the sensors employed (e.g. PIR sensors, contact sensors) provide low resolution information. Meanwhile, the expansion of ubiquitous computing allows researchers to gather additional information from different types of sensor which is possible to improve activity analysis. Based on the previous research about sitting posture detection, this research attempts to further analyses human sitting activity.

The aim of this research is to use non-intrusive low cost pressure sensor embedded chair system to recognize a subject's activity by using their detected postures. There are three steps for this research, the first step is to find a hardware solution for low cost sitting posture detection, second step is to find a suitable strategy of sitting posture detection and the last step is to correlate the time-ordered sitting posture sequences with sitting activity.

The author initiated a prototype type of sensing system called IntelliChair for sitting posture detection. Two experiments are proceeded in order to determine the hardware architecture of IntelliChair system. The prototype looks at the sensor selection and integration of various sensor and indicates the best for a low cost, non-intrusive system. Subsequently, this research implements signal process theory to explore the frequency feature of sitting posture, for the purpose of determining a suitable sampling rate for IntelliChair system.

For second and third step, ten subjects are recruited for the sitting posture data and sitting activity data collection. The former dataset is collected by asking subjects to perform certain pre-defined sitting postures on IntelliChair and it is used for posture recognition experiment. The latter dataset is collected by asking the subjects to perform their normal sitting activity routine on IntelliChair for four hours, and the dataset is used for activity modelling and recognition experiment. For the posture recognition experiment, two Support Vector Machine (SVM) based classifiers are trained (one for spine postures and the other one for leg postures), and their performance evaluated. Hidden Markov Model is utilized for sitting activity modelling and recognition in order to establish the selected sitting activities from sitting posture sequences.

After experimenting with possible sensors, Force Sensing Resistor (FSR) is selected as the pressure sensing unit for IntelliChair. Eight FSRs are mounted on the seat and back of a chair to gather haptic (i.e., touch-based) posture information. Furthermore, the research explores the possibility of using alternative non-intrusive sensing technology (i.e. vision based Kinect Sensor from Microsoft) and find out the Kinect sensor is not reliable for sitting posture detection due to the joint drifting problem. A suitable sampling rate for IntelliChair is determined according to the experiment result which is 6 Hz. The posture classification performance shows that the SVM based classifier is robust to “familiar” subject data (accuracy is 99.8% with spine postures and 99.9% with leg postures). When dealing with “unfamiliar” subject data, the accuracy is 80.7% for spine posture classification and 42.3% for leg posture classification. The result of activity recognition achieves 41.27% accuracy among four selected activities (i.e. relax, play game, working with PC and watching video).

The result of this thesis shows that different individual body characteristics and sitting habits influence both sitting posture and sitting activity recognition. In this case, it suggests that IntelliChair is suitable for individual usage but a training stage is required.

Table of Contents

List of Figures

List of Tables

List of Abbreviation

3D Three Dimensional

ADC Analog-to-Digital Converter

ADL Activity of Daily Living

AI Artificial Intelligence

AmI Ambient Intelligence

AUC Area Under Curve

BPMS Body Pressure Measurement System

BSN Body Sensor Network

CV Cross Validation

CASAS Center for Advanced Studies in Adaptive Systems

DAQ Data Acquisition

DFPS Distance from Posture Space

ECG Electrocardiogram

EMFi Electromechanical Film

EPMs Eigen Pressure Maps

FSR Force Sensing Resistor

FPS Frames Per-Second

GMM Gaussian Mixture Model

HCI Human Computer Interaction

HMM Hidden Markov Model

MCC Measurement Computing Corporation

MEMS Micro Electro Mechanical System

RFID Radio Frequency Identification

ROC Receiver Operating Characteristic

LDA Linear Discriminant Analysis

IDE Integrated Development Environment

IE Intelligent Environment

PCA Principal Component Analysis

SDK Software Development Kit

SVM Support Vector Machine

Chapter 1 Introduction

The research field of Ambient Intelligence (AmI) or Intelligent Environment (IE) can be traced back few years starting with the need of home automation (in early 1990s). The purpose of which is to establish the connection between the occupant and their environment around, so this concept could also be understood as a living or working space that interacts in a natural way and adapts to the occupant. The concept originate from the idea of Ubiquitous Computing (Weiser 1991), and further considers the users as the centre of the system with all devices physically integrated into the environment (Tscheligi et al. 2009). This idea can benefits a wide range of application domains including healthcare, public environment management, practical working or learning skill training support, etc.

Along with the technology development, there is a trend that AmI needs to reposition itself and extend its landscape, because the major challenge is move from how to embed AmI into real world into a stage that is more concerned with finding ambient solutions for real-life problems (Tscheligi et al. 2009). In 2013, the concepts which are related to this research areas like Ambient Intelligence and Intelligent Environment is expand itself with many other research disciplines like Human Computer Interaction (HCI), Pervasive/Ubiquitous computing. AmI and IE is also benefit from technologies like sensor and actuator technology, Artificial Intelligence (AI) and Networking and middleware which provides their foundation. The relations between those technologies and research areas are shown in Figure 1-1.

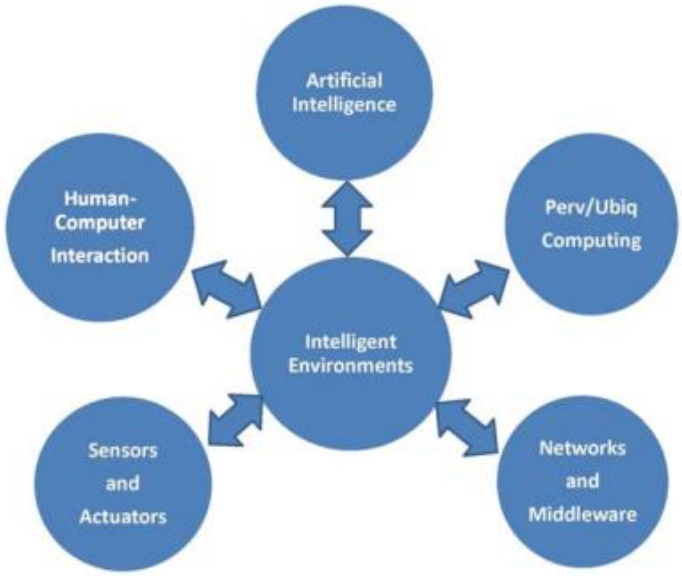


Figure 1-1 Interaction between the area of IE and other disciplines (Augusto et al. 2013).

Existing AmI or IE research, such as the CASAS (Center for Advanced Studies in Adaptive Systems) project (Cook et al. 2013, Cook et al. 2009) focuses on the interpretation of a subject’s behaviour at the activity level by logging the Activity of Daily Living (ADL) such as eating, cooking, drinking, and taking medicine. In general, the sensors employed (e.g. PIR sensors, contact sensors) provide low resolution information. Combined with the measurements of room temperature, humidity and other indoor environment information, intelligent environment systems are able to provide valuable functions like remote health monitoring and intervention. Meanwhile, the expansion of wearable computing allows researchers to attempt to improve human behaviour analysis by gathering additional information (Wang et al. 2012, Lo et al. 2005, Brady et al. 2006).

Compared with the activity detection method in Cook's Research (Cook et al. 2009), motion tracking technology provides more detail about human biomechanical movements and this additional information allows the system to detect activities more easily. Additionally, because of the AmI's multiple disciplines feature, the participation of more individual sensor systems is demanded, which explains the rise of wearable computing smart device market.

The purpose of those wearable computing smart devices (including smart watches, smart band, etc.) is to track the human daily activity based on the human motion or movement detection. Those devices provide additional information for human activity analysis, but they still have drawbacks. Firstly, those devices are part of the Body Sensor Network (BSN) (Lo and Yang 2005, Lo et al. 2005) or Body Area Network (BAN) (Latré et al. 2011) which those devices still require to be installed on human body. Secondly, the focus of those devices is the outdoor movement based activities (location tracking for running excesses, or walking distance calculation) or biomedical information (Blood pump in vein, ECG, etc.). Figure 1-2 shows the possible information that those smart device could collect.

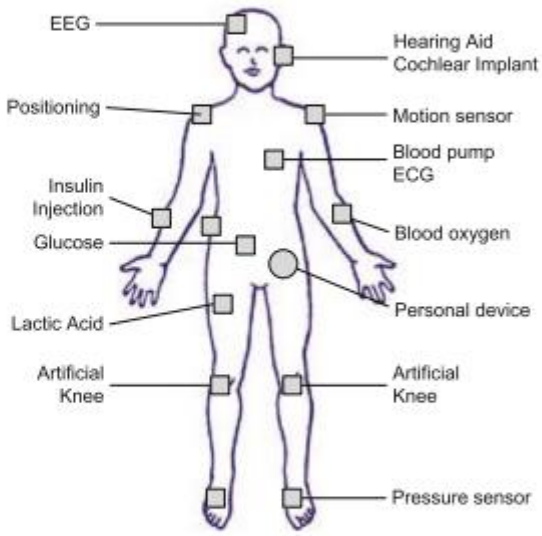


Figure 1-2 Example of patient monitoring in a Wireless Body Area Network (Latré et al. 2011).

Those drawbacks are the motivation to carry out this research which aims to make contributions in non-intrusive motion detection (sitting postures) and indoor activity (sitting activities) analysis. The IntelliChair is proposed as an approach, which combines posture classification accuracy with more detailed information of sitting activity. The goal of the system is to track the naturally occurring sitting postures of a user through the use of non-intrusive, low cost, chair surface-mounted sensors and establish a correspondence between **sitting postures** and **sitting activities** such as relaxing, watching TV, play games, etc.

The early research about the sitting posture is major focused biomedical topics. Tan's research (Tan, Slivovsky and Pentland 2001) considered as the starting point of early stage. In their work, pressure sensor array integrated mats are mounted on the chair surface and the system is able to classify different postures. Then the research has been expanded and crossed with other disciplines:

- Novel HCI approach: Using audio to monitor and represent the sitting posture change (Hermann and Koiva 2008).
- Affective computing: Detect the child's interest level when engaging with games through posture (Mota and Picard 2003).
- Stress level detection: Detect the stress level through pressure change of sitting posture (Arnrich et al. 2010).
- Sitting discomfort detection: monitoring the car driver's sitting discomfort (Hermann 2005).

Those research all consider their system as a standalone system to accomplish specific tasks. This is the advantages of IntelliChair; it could either run standalone for sitting posture recognition, furthermore, its outcome of activity recognition could be additional data source for other AmI systems. In this thesis, the content is concentrating on the IntelliChair system which covers the hardware design, sitting posture recognition and sitting activity recognition.

1.1 Aim and Objectives of the Research

As introduced above, the aim of this thesis is to build up a system that is capable of recognize sitting activity through a non-intrusive and low-cost sensor system.

According to the description in the introduction, the research objectives in this thesis are:

1. Develop a non-intrusive hardware system that is capable of collecting sensor data at a relatively small financial cost with high accuracy.
2. Detect the sitting posture through the collected data.
3. Establish a correspondence between sitting posture and sitting activities.

The first objective is to build up a system that is affordable, because the sensor systems that are utilized in the past research (those research projects is described in chapter 2) cost thousands of pounds. Thus, the choices of different type of sensors, how to deploy the sensors and the integration of different sensor technology will be explored in the thesis. The relevant issue, the sampling frequency of the system, will be discussed for it provides foundation for both posture and activity data collection. Furthermore, in order to make the system capable of accomplishing the posture detecting task on its own, so the most recent micro PC system will be integrated as a part of the IntelliChair.

The second objective investigates the use of using machine learning method to classify the input data into pre-defined sitting posture classes. The output of the posture classification forms the input to the sitting activity recognition component that is the third objective of the thesis. The activity modelling and recognition uses Hidden Markov Models (HMM) to correlate the posture sequence with several specific activities, because HMM is modelling method that is used to deal with sequential data, which is the posture sequence in this thesis.

Overall, there are five research questions that are extended from the objectives, which comprise:

1. (Objective 1a): What type of pressure sensor should the system use?
2. (Objective 1b): Are there any alternative sensors to achieve non-

intrusive detection other than pressure sensing?

3. (Objective 1c): What is the signal characteristic of the pressure sensing information?
4. (Objective 2): How to classify the data in order to detect the sitting posture?
5. (Objective 3): How to build up the correspondence between the users' sitting posture and the users' activity?

1.2 Outline of Thesis

The content of this thesis is organized according to the research questions and the relevant experiments. The thesis structure is as follows:

Chapter 2 (Literature Review): This is chapter for literature review part. It provides an overview of sitting posture research. It not only reviews the previous research and projects in human sitting posture detection following the tendency of sensor number reducing, but also reviews the related sensor technologies.

Chapter 3 (Methodology and Overall Experimental Design): This chapter design the methodology to the five research questions of this thesis based on the overview of chapter 2 and proposed the IntelliChair system. It also established five experiments; each of the experiment is correlated to one research question.

Chapter 4 (Experimental Results): This chapter describes the procedure of the first four experiments and explains the result in order to answer the proposed research questions. The first four experiments include: the selection of the pressure sensor, the integration of different types of sensors, the posture signal analysis and the posture classification using machine learning.

Chapter 5 (Result of Sitting Activity Modelling and Recognition): The content of this chapter focuses on the experiment for the activity modelling and recognition and explains the result of the experiment.

Chapter 6 (Conclusions and Future Work): This chapter includes overview, discussion and summary for this research which includes the conclusion and suggestions for future work.

Chapter 2 Literature Review

This chapter introduces and discusses the relevant research of sitting posture recognition systems, pressure sensor technology, alternative sensor technology and related work of middleware that informs the IntelliChair system.

This chapter provides the foundation of most relevant ideas in this thesis; furthermore, it leads to experimental design in chapter 3. More specific issues in research questions that are necessary for this thesis are introduced and discussed in each chapter.

This chapter emphasizes particularly the application of pressure sensing technologies in posture recognition. Because the purpose of the IntelliChair is to achieve non-intrusive posture sensing, so through the studies on the past research on posture recognition, the first concern is the sensor technologies that is used in that past research include but not limited in pressure sensing technology. Secondly, the algorithms and methods for posture recognition are address. Thirdly, how the training data is established and what element should be included is discussed.

2.1 Studies on Sitting Posture Recognition Systems

This section focuses on the discussion of previous research about sitting posture detection and recognition. The discussion consists of two parts:

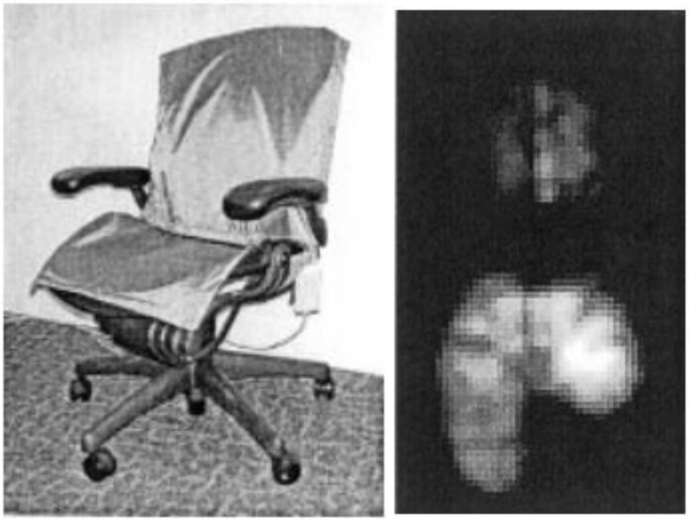
- The hardware architecture of those systems.
- The posture recognition strategy, algorithm and methods that are utilized in those systems.

The first part of the discussion includes the sensors used, and the sensor placement strategy. The main focus is the way to achieve the simplification of sensor system. The second part of the discussion covers related topics such as building of training and test data, and evaluates the system performance and the data analysis strategy for pressure based sitting posture recognition.

2.1.1 Sensing Chair Project

The Sensing Chair project (Tan, Slivovsky and Pentland 2001, Tan 1999) was launched by Haptic Interface Research Laboratory in Purdue University. The Sensing Chair project aims to solve the shortage of a research platform in the multimodal interface research. It is a real-time sitting posture classification system using a surface-mounted pressure sensing mat placed on the seat pan and backrest of a chair. The goal of the project is to build a robust multi-user sitting-posture tracking system with applications including ergonomics and automatic control of airbag deployment in a car. For example, a sensing driver's or passenger's seat can automatically adjust an airbag's deployment force according to the estimated weight and size of the driver, or disable the airbag if an infant car seat is detected in the front seat.

The pressure sensing mat utilized in the sensing chair project is the Body Pressure Measurement System (BPMS) manufactured by Tekscan Inc. (Tekscan 2015). The Chair system consists two identical surface-mounted pressure-sensitive transducer sheets, with PC interface board. Each ultra-thin sheet is printed with an array of 42-by-48 sensing units and measures 0.10 mm in thickness, and the units are uniformly spaced with a 10mm inter-element distance. As Figure 2-1 shows, when a force is applied on the surface, a real-time pressure map image for sitting posture is generated. The pressure map image represents the sitting posture at that time frame and the data analysis is based on the image processing.



(a) (b)

Figure 2-1 The sensing chair system (a) and a full pressure map (b).
The posture “left leg crossed” is show as a 2-D grayscale image in (b). The left side of the map corresponds to the right side of the occupant (Tan, Slivovsky and Pentland 2001).

The BPMS is a qualified pressure sensing system product because it not only quantifies the force applied on the surface but also the exactly the position of force. This is because of its high density array design of transducers (4032 transducers in total). Meanwhile, Tekscan, Inc. provides a set of toolkit software to support the data acquisition, recording and visualisation. Because of those features, The BPMS has been widely used in many theoretical research like stress level identification (Arnrich, Setz et al. 2010) or learner’s interest detection (Mota and Picard 2003) and so on.

However the price of the BMPS has limited its adoption, because the whole BMPS system cost thousands of pounds. So BMPS is unsuitable from the hardware selection of the IntelliChair project for one of the project goals is to construct a low-cost hardware system.

The first paper of the sitting posture research by using pressure sensor is published in 1999 (Tan 1999). In this paper, the Sensing Chair system is firstly introduced. The author introduced a real-time posture classification system (Sensing Chair system) by using surface-mounted pressure sensors which are placed on the seatpan and backrest of a Chair. The major concern of this paper is the classification approach which is based on the Eigen Pressure Maps (EPMs) and Distance from Posture Space (DFPS) (Tan 1999). The process of the approach is divided into two parts:

- Pre-process (training stage): This part is based on the collected training dataset. The raw pressure map is transformed to a mean-deviation based matrix and an eigenvector array for each posture built up this matrix. By combining the eigenvector arrays from each posture, the EPMs are established.
- Classification (prediction stage): When a new pressure distribution map is input, the DFPS use the mean-deviation difference as the entry to calculate the deviation between the input data with each individual posture EPM space where lowest value of the DFPS is found. The mean-deviation is the indicator that which classes the input pressure distribution map

should belong, because the lowest value means the input posture pressure data is most similar to this posture class.

The essence of this approach is the Principal Component Analysis (PCA) based classification. The reason to implement PCA is to reduce the dimensionality of the data (raw data has 4032 dimensions because of the 42-by-48 sensing units of the BPMS), and interpret it with the distribution of principal components of the pressure map. In this case, the PCA transformed the original pressure map from a 4032-dimensional space into a 10 dimensional space, because there are 10 pressure distribution maps for one posture. This is achieved by using the eigenvector concept in linear algebra, to build up the correspondence between the covariance matrix of AA^T (A is matrix of size 4032*10, so this matrix is size of 4032*4032) and $A^T A$ (this matrix size is 10*10). Through this transition process, the computation is dramatically reduced.

In this paper (Tan 1999), Tan defined 14 basic postures based on Lueduer's research (Lueder and Noro 1994). The 14 postures are: 1) seated upright; 2) leaning forward; 3) leaning left; 4) leaning right; 5) right leg crossed (with knees touching); 6) right leg crossed (with right foot on left knee); 7) left leg crossed (with knees touching); 8) left leg crossed (with left foot on right knee);

9) left foot on seatpan under right thigh; 10) right foot on seatpan under left thigh; 11) leaning left with right leg crossed; 12) leaning right with left leg crossed; 13) leaning back; and 14) slouching.

The following paper in 2001 (Tan, Slivovsky and Pentland 2001) further discussed the system detail especially the new added multi-user static posture classification system. There are several changes to the training data collection strategy including:

- Pre-defined sitting postures are reduced from 14 to 10. The 10 postures are: i) seated upright; ii) leaning forward; iii) leaning left; iv) learning right; v) right leg crossed; vi) left leg crossed; vii) leaning left with right leg crossed; viii) leaning right with left leg crossed; ix) leaning back; and x) slouching. “These postures are similar to the 14 postures utilized in the single-user system with two exceptions. First, the two types of leg crossing (one with knees touching, the other with a foot on the other knee) have been combined, thus postures 5) and 6) are now v), and postures 7) and 8) are now vi). Second, the two postures that require a subject to sit on a foot, namely 9) and10), have been eliminated because some subjects are unable to do so” (Tan, Slivovsky and Pentland 2001) .
- Subject number in which the training data is based was expanded to 30 people (15 females and 15 males) in order to cover a wide distribution of

anthropometric measurements.

- Each subject contributes five pressure maps for one posture instead of ten. It is aim to simplify the data collection process for the number of subjects has expanded.
- Eight out of thirty subjects' data is used for test data and their pressure distribution maps are not included in the training data, but used to validate the trained classifier. It is used to test the classification performance when the Sensing Chair deals with "unfamiliar" subject data.

In order to better understand the validation result for the classifier, the authors introduces the concept of subject similarity (Tan, Slivovsky and Pentland 2001). In the paper, it is called "familiar" subject and "new" subject. "Familiar" subject means a new subject in test data with similar body characteristics (such as weight) with one of the subject within the training data, and this means the pressure distribution map they generated may be very similar as well. Meanwhile, "new" subject means a new subject in test data is not covered by the training data subjects. For example, in the Sensing Chair project, the weight of the male subjects for training data covers between 146 to 260 lbs (Slivovsky and Tan 2000), if there is a new subject A whose weight is 200 lbs, the pressure distribution maps from subject A may be covered in training data, which means A is a "familiar" subject. If there is a new subject B whose weight is only 100 lbs, the pressure distribution maps from subject B may have a low similarity with the maps from training data, and it is called "new" subject.

The average accuracy is 79% when the trained system deals with "unfamiliar" subject, and the average accuracy is 96% when the system deals with "familiar" in the validation stage using test data as showed in Figure 2-2. According to the classification algorithm description in Tan's paper (Tan 1999) , the EFPS value represents similarity of two posture data. The lower the EFPS value, two posture data are more similar. In an effort to locate the sources of error, Tan examined the posture labels associated with not only the minimum DFPS (the first choice), but also those with the next two smallest EFPS values (the second and third choices). The curves in both graphs of Figure 2-2 indicates that similar performance levels with both "familiar" and new subject groups can achieves nineties percentage if the correct posture label can be derived from the first three smallest EFPS values. This results shows the importance of correlated posture label for classification and posture labelling should be a part of posture training data collection.

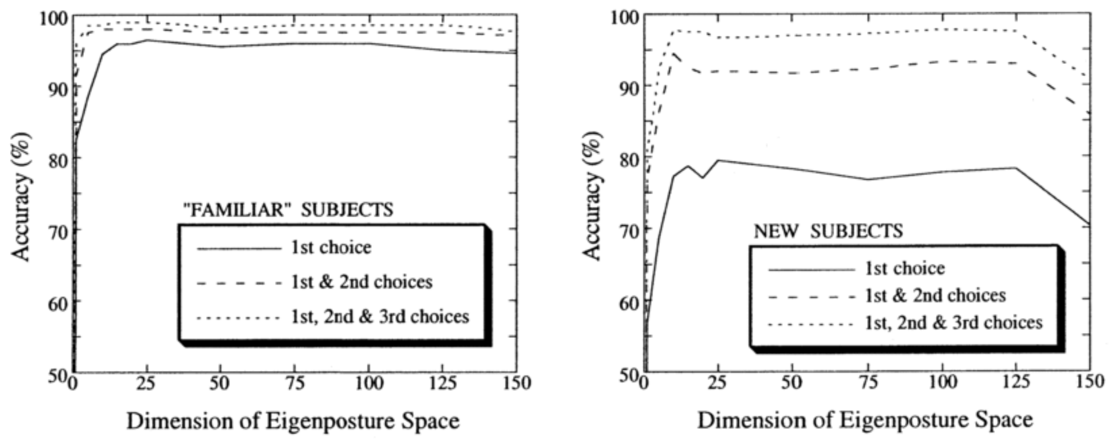


Figure 2-2 Classification result from Tan's research (Tan, Slivovsky and Pentland 2001). (Left) Classification accuracy that classifier deal with "familiar" subject. (Right) Accuracy that classifier deal with "unfamiliar" subjects. Solid, dashed and dotted lines correspond to accuracies associated with the smallest EFPS value, first two smallest EFPS values, and the first three smallest EFPS values, respectively.

This research is a fundamental base for the sitting posture analysis through pressure data which provides the foundation of IntelliChair project which are:

- Sitting posture detection is possible through pressure, and the sitting posture could be well represented.
- Individual differences significantly affect the classification performance. It means the evaluation of a pressure based system should include both "familiar" and "unfamiliar" test.
- The procedure to collect training data for a pressure sensing system should label the pressure data with the pre-defined sitting posture.

In additional, Slivovsky and Tan's research (Slivovsky and Tan 2000) record the subject's information (gender, age, weight and height) in order to shows that the data they collected covers a wide range of subjects. In this thesis, the additional information such as gender, age, weight and height of the subjects are recorded as well. The reason for including the information is because the subject's individual difference is potentially relevant to the sitting posture recognition performance, especially a recognition system with limited number of sensors. Compared to the posture data from Tan's Sensing Chair which is high resolution data (with 4032 sensors), limited sensor system generates low resolution data from which it is possible to emphasis the posture difference that might be caused by individual physical differences.

2.1.2 Reduction of Sensor Numbers

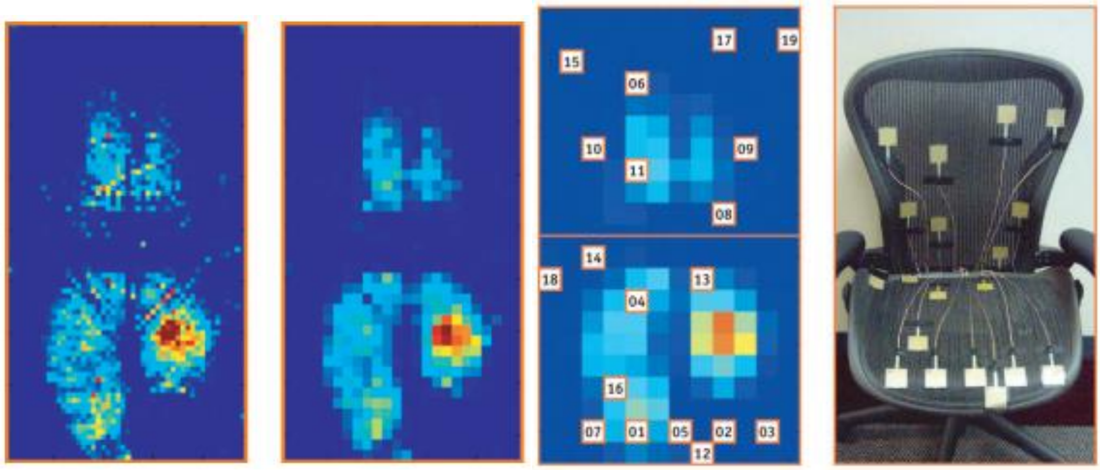
Since Sensing Chair project has set the foundation of the sitting posture recognition research area, some researchers are motivated by the significant potential for posture recognition with haptic interface. There are researchers who have expanded the research based on the hardware platform of Sensing Chair. The research group of Tessendorf (Tessendorf et al. 2009) aims to improve the performance in an unsupervised manner. Another group from MIT Media Lab (Mota and Picard 2003) implements the Sensing Chair to associate the sitting posture with subject's interest level of learning. However, the hardware cost is the limitation for large scale usage for all those systems. Therefore, some researchers are also looking to detecting the posture through pressure in an in-expensive way, which is the goal of the IntelliChair project.

Also, a sensing system with large number of sensors means it could generate high-dimensional data. High-dimensional data could benefit the resolution of the data in representing phenomenon, but it has negative effect on data analysis because it will cause the problem which is known as the “curse of dimensions” (Du 2010, pp. 7). It means “when data are scattered, no patterns converge in the high-dimensional space”. In other words, the least relevant and least influential dimensions should be ignored, while the most relevant and significant dimensions is considered. In the Sensing Chair project, the classification methods used by the researchers are based on dimension reduction (like PCA).

The meaning of sensor reduction is based on the consideration of hardware cost and simplifying the system, and more importantly, fewer sensors will physically lower sample data dimensions and size. The research group (Mutlu et al. 2007) from Carnegie Mellon University leads the way to reduce cost of sensors by replacing the Tekscan BPMS (total 4032 sensors) into a set of 19 sensors which is shown in Figure 2-3. Their work has drastically reduced the hardware and computational cost.

Figure 2-3 The sensor placement from Mutlu's research (Mutlu et al. 2007).
Left is the Sensing Chair with Tekscan, and the one on the right is from research group of Carnegie Mellon University.

In order to achieve the sensor reduction, Mutlu (Mutlu et al. 2007) use the feature selection method to obtain the most significant dimensions based on Tan's pressure distribution map approach (Tan, Slivovsky and Pentland 2001) as shown in Figure 2-4. The reason for the utilization of the feature selection is because the authors (Mutlu et al. 2007) consider that where pressure is applied is more informative about user's posture than the amount of pressure applied on the surface.



(a) (b) (c) (d)

Figure 2-4 Dimension decrease process of Mutlu's research. (a) is the original pressure map from Tan's research, (b) is the pressure map after blur process, (c) is the outlined 19 factors by the feature selection process, (d) sensor placement on Chair based on the feature selection process.

With this consideration, the original pressure map (Figure 2-4(a)) from Tan's research (Slivovsky and Tan 2000) is first transformed into a low-resolution pressure map (Figure 2-4(b)). Then a Support Vector Machine (SVM) based feature selection algorithm is implemented to find the most significant factors or features in low-resolution pressure map (Figure 2-4(c)). After the feature selection process, 19 factors are located on the pressure map, and Figure 2-4(d) is the re-constructed Chair system by using FSR sensor instead of Tekscan BMPS system. All 19 FSR sensors are deployed based on the placement in Figure 2-4(c).

Why choose 19 as the feature numbers? In order to determine the best feature numbers, Mutlu research group conducted a simulation experiment. This experiment uses the original pressure map as a base, and simulates a range number (from 1 to 4032) of features which is a repeat process of feature selection which is similar in Figure 2-4(c). The output of the experiment is the correspondence of the simulated sensor numbers (or feature numbers) and their classification result. Figure 2-5 shows classification result along with the increasing number of simulated sensors.

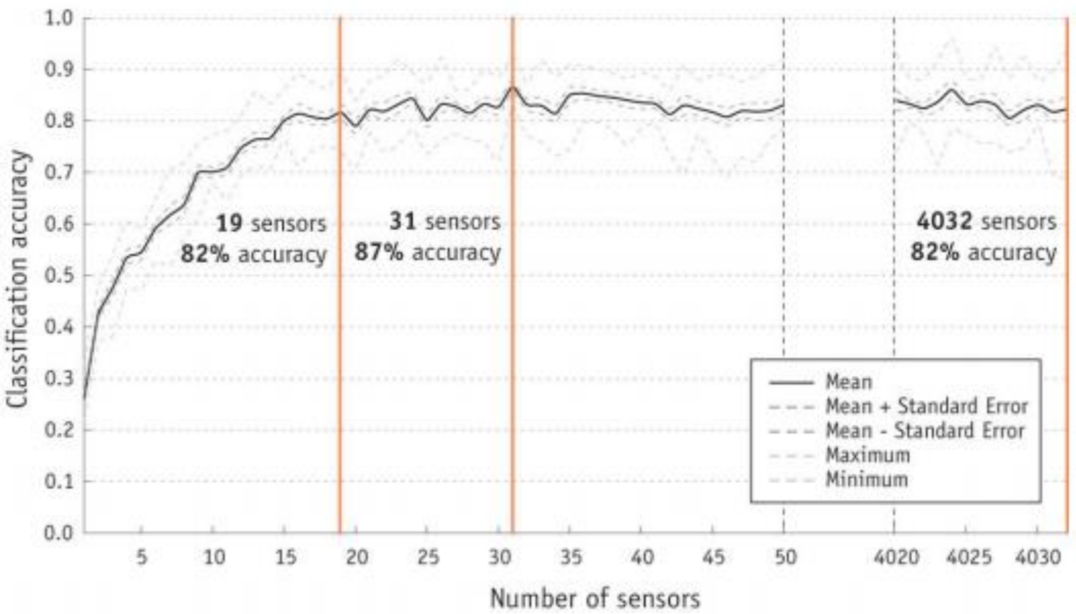


Figure 2-5 The correspondence between the simulated sensor numbers (selected features) and the classification accuracy (Mutlu et al. 2007).

Figure 2-5 indicates that 31 simulated sensors have the best mean accuracy at 87%. But 19 simulated sensors which have relatively lower mean accuracy at 82% are selected for system design because of the lower hardware cost.

Additionally, the Logistic Regression is selected as the classification technique which is different from Tan's research (Tan, Slivovsky and Pentland 2001).

However, one important thing is that the performance of 87% and 82% is based on the reconstruction of Tan's data (Tan, Slivovsky and Pentland 2001). Mutlu's group performed their own experiment with 20 subjects with 10 pre-defined sitting postures (the same postures with Tan's research). The final result of Mutlu's research achieves 78% accuracy with 20 subjects by using 19 sensors.

Because the sensor number reduction and their placement optimisation is achieved through the analysis of original pressure distribution maps from Sensing Chair project, in order to evaluate the new pressure sensing system, the authors of this research reconstruct the pressure distribution maps, for validation purposes. Figure 2-4 also shows the reconstruction process, by which a blur process converts the original pressure distribution map image into a low resolution image. This blur process concentrates several pixels from the original image into one pixel. By matching the new sensor locations with the blurred image, the pressure measurement for new sensor is speculated.

Mutlu's work (Mutlu et al. 2007) firstly use the data collected from 19 sensor strategy as the classifier training data, then use the reconstructed data from Sensing Chair project to evaluate the classifier, and the final result was only 63%. On the other hand, if the evaluation data is from the 19 sensor strategy, the accuracy is 78%. The authors (Mutlu et al. 2007) conclude that the difference is because of the quite different and lower-fidelity signal response between the sensor Mutlu used and the BPMS from Tekscan, Inc as well as variance of the subjects' posture. This finding indicates that the unification of pressure sensor is another important point that needs to be considered in sensor system design.

The major contribution of Mutlu's paper (Mutlu et al. 2007) is that it further clarifies the pressure based sitting posture data. It indicates the following relation between sensor location and amount of pressure:

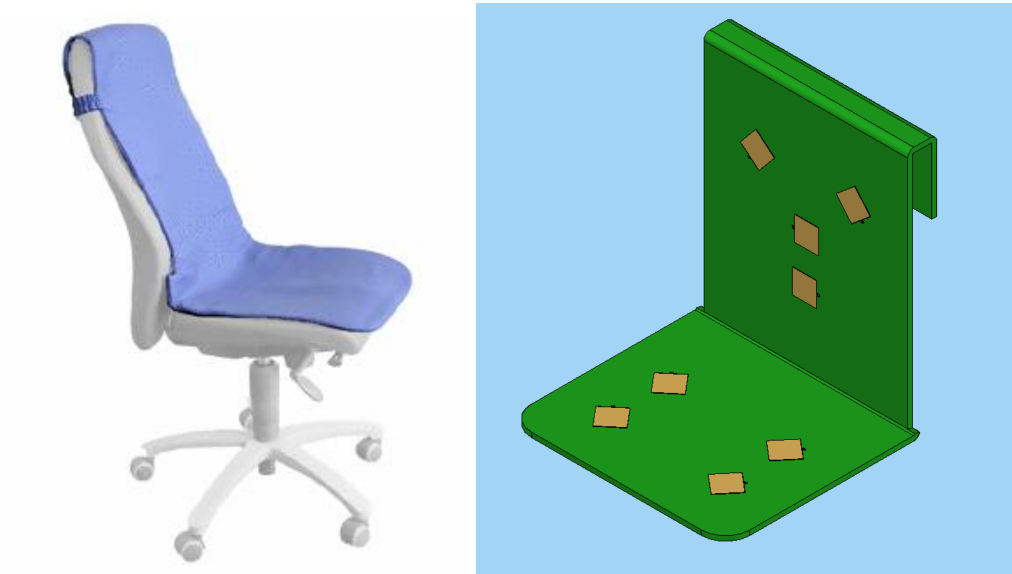
- While dealing with multiple subjects, high-resolution sensor data will leads to curse of dimensionality problem. This problem will lead to overfitting on classifier, thus, the classification accuracy on unfamiliar data is significantly lower compared to familiar data. So low-resolution data might be a better approach for multi-subject posture classification because of the sensors is placed at critical positions on the chair surface.
- Sensor deployed at correct position is more meaningful than how much

pressure the sensor measured. This is a very important finding which is the motivation for further reduction of sensor numbers.

- Different types of pressure sensors have different sensing mechanisms, so the unification of sensor needs to be considered. This means that the numerical conversion between pressure and sensor reading is different, so the same pressure will generate different reading by different sensor. This finding is important for the pressure sensing system design. It means when designing a pressure sensing system, the sensors within one system should have the same specification, which make sure the collected data has a uniformed signal response. Furthermore, the data collected by one system should not be used to evaluate another system; otherwise the classification performance will not be ideal.

2.1.3 Further Reduction of Sensor Numbers

The tacTiles mat project (Hermann and Koiva 2008) is inspired by the concept of Cognitive Interaction Technology in HCI. It aims to provide a novel way of interaction using the tactile sensitive surface as the interface for human-computer interaction or ambient information system. Herman and Koiva (Hermann and Koiva 2008) described the structure of the tacTiles mat hardware and show its applications that demonstrate real-time sonification (a research field which aims to translate given information of any kind into auditory feedback to user) for sitting monitoring and biofeedback.



(a) (b)

Figure 2-6 The tacTile mat system lay-on chair and the position of the sensors (Hermann and Koiva 2008). (a) The tacTile is deployed on office chair. (b) The sensor placement within the tacTile mat.

Figure 2-6 shows the sensor placement on the mat surface in order to collect the tactile information when the subject is sitting on the mat surface. The type of sensor used is called Force Sensing Resistor (FSR) which is also the sensor used in Mutlu's research (Mutlu et al. 2007). The sensor will measure the tactile information from the sensors, and the data is transmitted to the host PC through Bluetooth communication. FSR is one of the pressure sensor options in this thesis, and its sensing mechanism is discussed in section 2.2.1 and its experiment performance is described in section 4.1.2.

The focus of Herman and Koiva's research (Hermann and Koiva 2008) is the translation from tactile/pressure information to sonification. Along with this objective, this paper utilise PCA method on pressure data and thus find the most meaningful features in order to map the pressure data into different characteristic sounds which allows the user to identify the pattern through the auditory signal. Herman and Koiva did not use classification method on the pressure data because they only need to identify the change between clusters instead of certain pre-defined classes because the auditory signal will become intense if the sitting posture of the tacTile mat user stay the same for a period of time (in order to prevent the sitting fatigue). So there is no accuracy evaluation result for the sensing system, but through the personal communication and discussion with Hermann and the tactile application videos (Hermann 2006), it shows the tacTiles mat system is capable of sitting posture sensing and it also shows that the FSR sensor unit is capable of accurate force sensing.

Furthermore, the sensor placement of tacTile Chair is more understandable compare with the placement strategy of Mutlu's research (Mutlu et al. 2007), because it is easy to convey the critical pressure generation position between human body and the chair surface. The sensor placement strategy of the tacTiles could be supported by the finding from another earlier research group from MIT Media Laboratory (Mota and Picard 2003). Mota and Picard’s research group utilised Tekscan BMPS system for pressure sensing, which is the same system of the Sensing Chair project.

In order to simplify the pressure distribution maps with high number of dimensions, Mota and Picard’s group use Gaussian Mixture Model (GMM) method to abstract the most effective features factors in the pressure distribution image (different from the SVM feature selection approach of Mutlu’s research group). Mota and Picard simplified the pressure distribution map with four clusters within the three - dimensional space (prior probability, mean and variance which are the parameters of GMM) of the seat surface (as showed in Figure 2-7) with another four clusters in the back surface (No figure for back surface is illustrated in the Mota and Picard’s paper). Figure 2-7 shows the GMM modelling on pressure distribution map. In the figure, each circle represents the range of variance parameter, and the mark inside the circle represents the mean parameter of one Gaussian distribution.

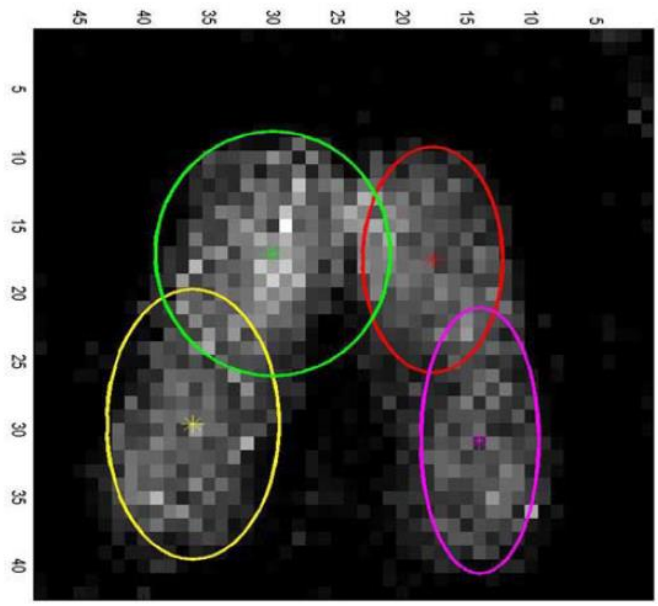


Figure 2-7 Seat pressure distribution matrix modelled with 4 Gaussians (Mota and Picard 2003).

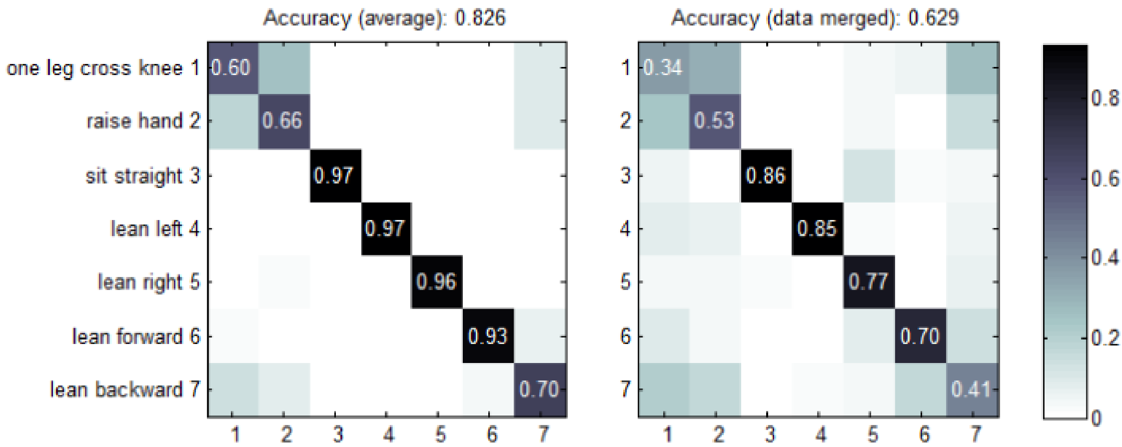
The Gaussian number of four that is chosen by Mota and Picard explains the geometric property of posture pressure pattern in the pressure distribution map (Mota and Picard 2003). Because the four clusters represent the most significant feature of the pressure distribution, it means that if there is one sensor is placed at the mean position of a Gaussian model, it is possible to use the measurement of one sensor to represent the pressure distribution of the area that this Gaussian model covers. Hermann’s tacTile mat (Hermann and Koiva 2008) utilized this sensor placement strategy and according to tacTile mat’s posture changing detection performance, the four sensor placement is validated to be reasonable.

After the simplification of data, Mota and Picard utilise Neural Network as the classifier to detect nine pre-defined sitting postures (sitting on the edge, leaning forward, leaning forward right, leaning forward left, sitting upright, leaning back, leaning back right, leaning back left and slumping back) (Mota and Picard 2003). In Mota and Picard’s research, there are 10 subjects and 5 subject’s data are used as test data. The overall accuracy reaches 87.64% when dealing with test data.

A project named Smart Chair (Cheng et al. 2013) has further simplified the sitting posture sensing system by using only 4 pressure sensors. But they are not deployed on the seat surface; instead, sensors are installed at the bottom of the chair as showed in Figure 2-8. Figure 2-8 (a) shows the pressure sensing material called conductive foam. The mechanism of this pressure sensing material is described in section 2.2.1 and the result of the experiment on conductive foam is discussed in section 4.1.1. Figure 2-8 (b) is the conductive foam based pressure sensing unit and in (c) is the position of sensor placement.

(a) (b) (c)

Figure 2-8 The pressure sensing unit and Smart Chair system (Cheng et al. 2013).



(a) (b)

Figure 2-9 The confusion matrix of all postures cross all subjects' data (Cheng et al. 2013).
(a) Average of all subjects.
(b) All data is merged into a single dataset.

As Figure 2-9 shows, the posture classification accuracy of Smart Chair is claimed to be 82.6% for 5 subjects with 7 posture classes by using Linear Discriminant Analysis (LDA) classifier (Scholkopft and Mullert 1999). The postures are sitting straight, leaning forward / backward / left / right, sitting with one leg cross the other knee, and sitting with one hand raised in the air (Cheng et al. 2013).

Figure 2-9 also shows that the classifier of Smart Chair has good performance for certain sitting posture like sit straight (97%), lean left (97%), lean right (96%) and lean forward (93%). The classification accuracy for other classes like one leg cross (60%) lean backward (70%) and raise hand (66%) are significantly poorer than previous four classes. Additionally, if the subject's posture data are merged, the overall classification performance decreased significantly as showed in Figure 2-9 (b).

Firstly, the reason for the accuracy difference is that the four sitting postures (sit straight, lean left, lean right and lean forward) have dramatic difference on pressure distribution with sensor under chair foot. Secondly, the reason for the poor performance when subject data is merged might be because the interference between different subjects. This is the main disadvantage of such sensor deployment because it is only sensitive to certain postures.

2.1.4 Summary of Studies on Sitting Posture Recognition Systems

According to the discussion in section 2.1, the decreased number of sensors not only reduces the cost of the hardware and simplified the sensor system, but also able to achieve acceptable sitting posture classification accuracy as showed in Table 2-1 (in the end of this section).

But the decreased number of sensors also leads to a decrease in recognition performance, so there is a decision to be made between the balance of sensor number and recognition performance. The sensor placement strategy of tacTiles mat project (Hermann and Koiva 2008) is selected for the IntelliChair system proposed in this thesis. The reason for this selection is based on:

- Sensor deployment: Hermann and Koiva's strategy is to deploy the sensors on the surface that the human body is directly contacted. This is potentially more reliable compared with Cheng's strategy (Cheng et al. 2013).
- Sensor Numbers: There are eight sensors on two surfaces (horizontal surface and vertical surface), and the system is able to detect the posture of the different body parts (torso and thigh). It is possible that

the separation of posture detection of different body parts could describe the sitting posture more precisely.

It is learned from the previous research that in order to build up a reasonable training dataset for sitting posture recognition model or algorithm, the data collection should cover multiple subjects (to test the generalization capability of the posture recognition system) along with the subject's individual characteristics (weight, height, etc.) because it is also a factor for recognition performance (Slivovsky and Tan 2000, Tan 1999, Tan, Slivovsky and Pentland 2001). It is also learned that in the hardware design stage, only one specific sensor product should be used in one sensing system (for example, within the pressure sensing system, do not mix two sensors that are from two different companies (Mutlu et al. 2007), even if their sensing mechanism is similar), otherwise the posture recognition performance is not ideal.

The discussion above provides the foundation for the experimental design which is described in chapter 3.

<table>
<tr><th>Authors</th><th>Year</th><th>Project Name</th><th>Number of Sensors</th><th>Number of Subjects</th><th>Sensor Location</th><th>Classification Algorithm</th><th>Classification Accuracy</th><th>Sensor Type</th></tr>
<tr><td>Tan, Slivovsky and Pentland</td><td>1999, 2000, 2001</td><td rowspan="2">Sensing Chair</td><td rowspan="2">4032</td><td>30</td><td rowspan="2">As showed in Figure 2-1, two BPMS, one on the vertical surface, one on the horizontal surface.</td><td>PCA based classification</td><td>Ten postures, 79% with “unfamiliar” subject, 96% with “familiar” subject.</td><td rowspan="2">BMPS system from Tekscan, Inc.</td></tr>
<tr><td>Mota and Picard</td><td>2003</td><td>10</td><td>Use GMM for feature exaction, after dimension deduction, Neural Network is used for posture classification.</td><td>87.64% with 9 postures</td></tr>
<tr><td>Mutlu et al</td><td>2007</td><td>Low-cost, non-intrusive seated posture recognition.</td><td>19 (based on the feature selection result)</td><td>20</td><td>As showed in Figure 2-3.</td><td>Logistic Regression (on reconstructed data). Simple Logistic, Naive Bayes, Support Vector Machine and Multi-Layer Perceptron for their own collected data.</td><td>Ten postures, accuracy are 87% on reconstructed data of Tan’ research, 78% with their collect data.</td><td>FSR</td></tr>
<tr><td>Hermann and Koiva</td><td>2008</td><td>tacTile mat</td><td>8</td><td>N/A</td><td>As showed in Figure 2-6, 4 on the vertical surface and 4 on the horizontal surface.</td><td>PCA (for clustering)</td><td>N/A, it is a monitoring system instead of a recognition system.</td><td>FSR</td></tr>
<tr><td>Cheng et al</td><td>2013</td><td>Smart Chair</td><td>4</td><td>5</td><td>As showed in Figure 2-8, sensors are on the floor, at the bottom of chair foot.</td><td>LDA</td><td>82.6% with 7 postures</td><td>Conductive foam based sensor</td></tr>
</table>

Table 2-1 Summary of different sitting posture recognition research.

2.2 Studies on Sensor Technologies

In this section, the sensor technologies that could be potentially used by IntelliChair is introduced. According the discussion in section 2.1, the major discussion in this section is pressure sensing technology. Additionally, several other sensor technologies (inertial sensor, vision based sensor) for motion and posture detection are discussed as an alternative option for pressure sensor.

A pressure sensor can be defined as a transducer or device that captures a signal or stimulus of the force applied, and converts it into a measurable electric signal. The output signal could be current, charge or voltage. Following the basic principle, there are many different approaches, the difference between them is the conductive material they use or the way they construct and organise the material.

From section 2.2.1 to section 2.2.3, different pressure sensing technologies will be discussed, along with the pressure sensing systems that refer to them. The purpose of the discussion is to find out the suitable sensor or sensing material to fulfil the pressure sensing task based on their sensing performance, reliability and cost. The pressure sensing technology that is discussed in this section includes conductive polymer, conductive fabric and optical fibre based. Those types of pressure sensing do not cover the whole pressure sensing area, but they are selected for they are widely used in Human Computer Interaction area. In section 2.2.4, alternative sensor technologies are discussed and the purpose of the discussion is to find out an alternative sensor technology other than pressure sensor to fulfil the non-intrusive posture detection task.

2.2.1 Conductive Polymer based pressure sensing

The first approach is the conductive polymer based pressure sensing technique (Brady, Diamond and Lau 2005, Lekkala and Paajanen 1999). The essence of this technique relies on polyurethane foam which is coated with the inherently conducting polypyrrole solvent. The current conductivity of this conductive foam is sensitive to the loaded force, which could be used as an indicator of the changing force (Dunne et al. 2005, Brady et al. 2006).

One type of conductive foam is called Smart Foam (Brady, Diamond and Lau 2005) and it is proposed for in cloth wearable computing purposes, by Sarah Brady's group from Dublin City University in 2005. This group identified the problem of medical devices for bio-information sensing (heart beat) is being invasive, uncomfortable to wear and requiring clinical infrastructure (some ECG detection devices require station machine to connect to the sensors and operating those machine requires clinical experience) to operate. Under the consideration that the body sensor device should not sacrifice the user's physical or social comfort, the Brady group proposed the novel Smart Foam based body sensing system that is able to be integrated into clothing, include the traditional electronic device properties like durability, low power consumption, and capable to connect into a circuit.

Brady's group then tailors the foam into a torso garment which is used to collect the breath rate data by monitoring the force changing of the ribcage area on the human body as showed in Figure 2-10 (Dunne et al. 2005).

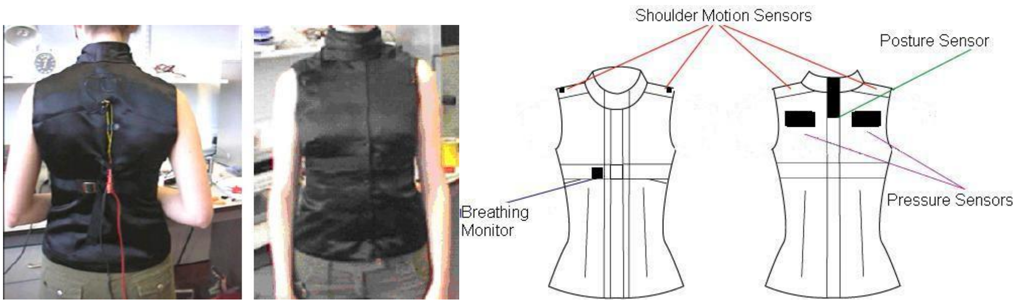


Figure 2-10 The prototype pressure-sensitive torso garment and the sensor layout (Brady, Diamond and Lau 2005).

In 2006, Brady's group compared the result from both novel Smart Foam sensor systems with the traditional medical Vmax metabolic system (a widely used clinical ECG monitoring system). The result shows the novel system is viable and easier for users to wear (Brady et al. 2006). The purpose of non-invasive pressure sensing is also one objective of this thesis. Since the Smart Foam pressure sensing material can be integrated into cloth, it is also possible to deploy the Smart Foam on a chair surface to measure the pressure.

Another type of pressure sensing material called Electromechanical Film (EMFi) is also utilized in this area (Lekkala and Paajanen 1999). This type of material is also sensitive to dynamic force applied on its surface. The difference between EMFi and the Smart Foam is their structure. The EMFi film has two permanent charge coat layers and a voided cellular polypropylene film between them as a sandwich structure. When force is applied, the air void in the middle layer is compressed and the charge between two coat layers is changed (Lekkala and Paajanen 1999).

The difference between EMFi and Smart Foam is that Smart Foam is capable of detecting the force in all directions whereas the EMFi film is only sensitive to vertically applied force and it is very in-sensitive to lateral forces.

EMFi is also utilised within the sitting pressure sensing research field. The research group from Tampere University has developed both wired and wireless EMFi sensor integrated chair to monitor the heart rate (Anttonen and Surakka 2005). The focus of Anttonen and Surakka research is the emotion and heart rate, and they find that EMFi chair could detect the significant difference between positive and negative emotions through the collected heart rate signal. Their continued work makes the EMFi chair system more reliable. However, this comes at the cost of disadvantage which is mainly difficulties in calibrating the system due to the complex amplifier circuit design (Junnila, Akhbardeh and Värri 2009).

One disadvantage for all conductive foam type of pressure sensing is that the foam cannot accurately locate the contact area; no matter wherever the area is large or small. It is because when the rubber polymer is compressed, the whole conductivity is changed accordingly, so the entire foam is considered as a whole unit.

The conductive polymer based system also has other applications in sensor and actuator research area. One of discipline is the organic transistor, and their application is in the robotic technology, which is called artificial skin.

A research group from the University of Tokyo first released a prototype of a large area organic transistor system (Someya et al. 2004) which is capable of both pressure sensing and thermal sensing. Another group from university of Cagliari of Italy proposed their organic transistor for pressure sensing with the potential to fit in wider application usage (Manunza and Bonfiglio 2007).

Apart from compact structure of the micro-electrode matrix design, the principle of the system is relying on the change of conductivity property of conductive rubber polymer while pressure is loaded. This matrix structure grants the sensor system the ability to track the position of the pressure contact area. But it also brings the drawback for the sensor system which is the bending diameter limitation (the matrix structure is normally designed as flat surface, but when the flat surface is bending into curve, the system still works but if the bending curve go over a certain diameter, the system is not able to work). The organic transistor is still classified as a microelectronic device, because tensile failure can occur. It means although its structure has certain flexibility, if the bending diameter is long enough, like 30mm (Someya et al. 2004), tensile failure occurs and the sensor structure will be damaged. The most recent sensor system that takes this concept in the sitting posture monitoring area is the group from University of Pavia, for the purpose of driving discomfort state monitoring (Marenzi et al. 2012). This matrix sensing approach is not include in this thesis due to the complex sensor design and the limitations in deployment, especially the thermal sensing function which is not essential for this research.

Another sensor option based on conductive polymer is the Force Sensing Resistor (FSR). It is also a conductive polymer based device which exhibits a decrease in resistance with and an increase in the force applied to the active surface (Interlink Electronics 2006). With an Analog-to-Digital Converter (ADC) support circuit, the sensor can quantify the applied force into an electric signal which could be measured. Several sitting posture recognition systems that is discussed in section 2.1 utilise FSR such as tacTiles mat project (Hermann and Koiva 2008) and Mutlu's research (Mutlu et al. 2007).

2.2.2 Conductive Fabric based Pressure Sensing

The conductive fabric is also named conductive textile and it is a fabric or nylon which has electric conductivity with the advantage of silk-like flexibility. However, conductive fabric is a very general definition of this material for the performance and usability depends on the way of manufacturing.

In the SenseChair (different from the Sensing Chair) project (Forlizzi et al. 2005), the conductive fabric is deployed on the chair on certain positions only to detect what parts of the chair the user is touching. It only generates binary signals in this case. Meanwhile, combined with complex amplifier and filter circuit design, conductive fabric based sensor could also acquire heartbeat and respiratory cycle information (such as ECG) (Choi and Jiang 2006). But it requires that the sensor is deployed tight on human body, in Choi's research, the sensor system is integrated into a belt to achieve the performance.

In this thesis, a sensor that can generate a numeric signal is preferable to one that can only generate a binary signal. This is because in the sitting posture detection scenario with sensor on the critical position, a numeric signal based sensor could detect the different level of contact pressure, while a binary signal based sensor can only detect whether this position is contacted or not. When the data is input into machine learning algorithms (i.e. Support Vector Machine), the low numeric signal would make less impact on the classifier compare with binary signal, thus, cause the posture classification is more accurate which is discussed in section 4.4. Although the conductive fabric could also generate numeric signal, it relies on the tight contact with human body, which becomes potentially intrusive.

2.2.3 Optical Fibre based pressure sensing

The concept of using optical fibre for pressure sensing is based on the light loss phenomenon. When optical fibre is bent in a micro radius curvature, the intensity of the light is decreased between the light input and light output.

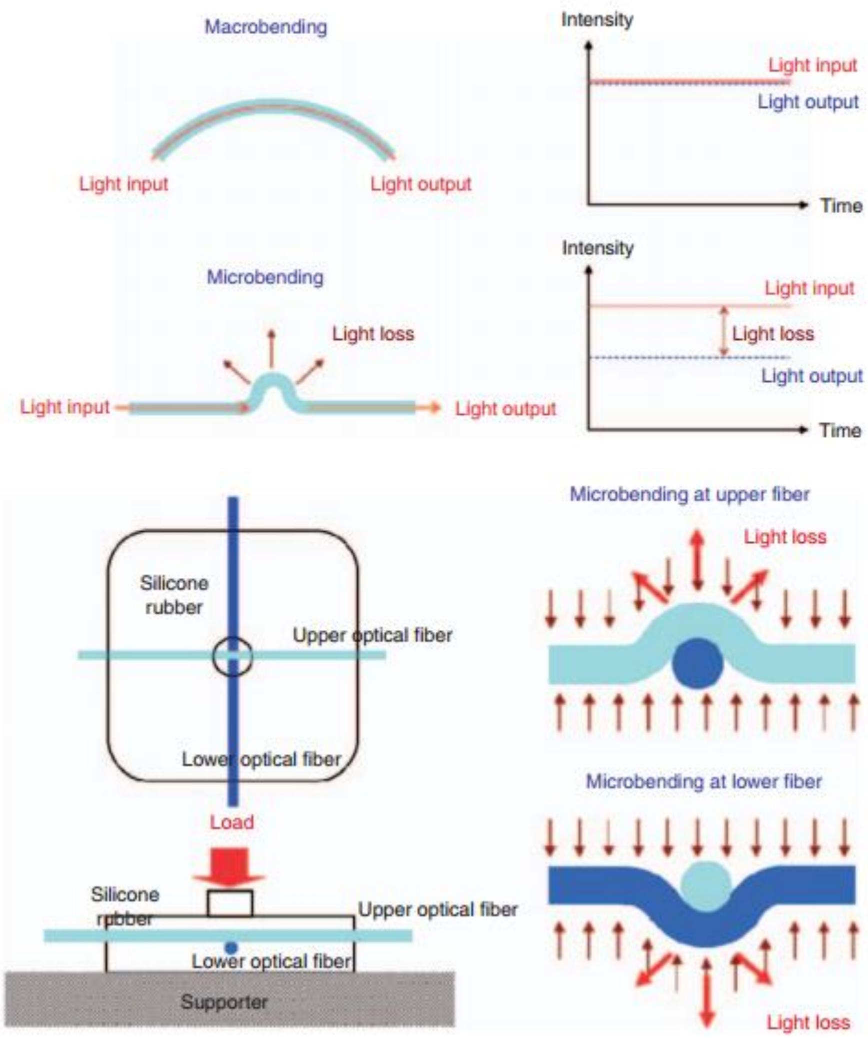


Figure 2-11 The effect of light loss in optical fibre micro bending situation and the sensor structure using fibre node (Heo, Kim and Lee 2009).

Figure 2-11 demonstrates the mechanism of optical fibre based pressure sensing. When two lines of optical fibre vertical cross against each other (two lines are contacted) up and down, there will be a node on their contact point. Then, the fibres are covered on both sides with silicone rubber (Heo, Kim and Lee 2009) or wove them into a cotton fibre structure (Rothmaier, Luong and Clemens 2008).

While pressure is loading on the node, the two fibres will micro bend each other (as showed in Figure 2-11), and the light loss phenomenon occurs. By measuring the light intensity loss between input and output, the amount of pressure can be measured. Furthermore, if multiple fibres are lined as a matrix, the sensor system could not only measure the amount of pressure, but also determine the position where pressure is applied. The optical fibre pressure sensing systems have the advantage such as high flexibility, being insensitive to electromagnetic radiation and not susceptible to electrical discharges, because they rely on the LED light source and coupled light detector device.

The drawback for this type of system is that it could only detect light force on sensing area (Heo, Kim and Lee 2009). Compared with former two sensing techniques (conductive foam and conductive fabric), the optical fibre based sensor is even more difficult to manufacture and relatively more expensive.

2.2.4 Alternative Sensing Technologies for Motion Sensing

With the improvement of Micro Electro Mechanical System (MEMS), inertial sensor based human motion sensing plays an important role in physical activity monitoring because of their portability, performance and cost (Yang and Li 2012).

There are many different types of inertial sensor; the two main types are accelerometer measurement based and gyroscope measurement based (Yang and Li 2012, Zeng and Zhao 2011). Accelerometer is a type of device that measures acceleration, including that induced by gravity, and gyroscope measures angular velocity. According to the previous research (Yang and Li 2012, Stirling et al. 2011), the general inertial sensor based human motion detection method demands the inertial sensor (accelerometer and/or gyroscope) to be attached to different parts of the user. Then, inertial sensors measure the acceleration and/or angular velocities which contain information related to human motion. Thus, motion estimation algorithms can extract the information from the collected sensor measurement.

According to the discussion above, there are mainly two reasons that the inertial sensor based sensing is not involved in this thesis. First, the inertial sensor needs to be attached on human body. This feature does not suits the aim of this thesis, which requires non-intrusive sampling for human body. Second, the information that is measured by inertial sensor is generated by the movement of human body (or body parts), such as the walking speed estimation (Yang and Li 2012), fall detection (Zeng and Zhao 2011) and sport medicine (Stirling et al. 2011). Refer to the aim of this thesis; the sitting posture recognition, where most of the time in the sitting situations, the human body keeps still, so the pressure sensing based method is more suitable.

Apart from the inertial sensing technologies, the vision based motion or posture sensing is another active research area. According to Moeslund, Hilton and Krüger's survey (Moeslund, Hilton and Krüger 2006), the vision based motion sensing technologies is able to perform tasks such as pose estimation, human presence and position detection and specific activity (carrying box across the room, etc.) recognition. Among those applications, the methods for pose estimation are another way to achieve non-intrusive posture detection. Within the pose estimation approaches, there are mainly two techniques, and they are:

- Use reflective markers on the human body to indicate specific body part in order to reconstruct the human body model (Ciampone 2012, Obdrzalek et al. 2012).
- Marker less approach: applied image process methods on the image data that is collected from single or multiple cameras to reconstruct the human body model (Corazza et al. 2006, Obdrzalek et al. 2012).

Referring again to the aim of this thesis again, non-intrusive posture detection is the objective, so the marker less vision based posture sensing is potential an alternative method for pressure sensing despite it might cause privacy obtrusive issue. The Microsoft Kinect sensor (Microsoft 2015) is an innovative device to achieve this marker less vision based posture sensing.

The Microsoft Kinect sensor (Microsoft 2015) currently draws a lot of attention in the vision based motion or posture detection area because it was primarily designed for natural interaction between the video game players and XBOX. The Kinect sensor captures *depth* and colour images simultaneously at a frame rate around 30 Frame Per-Second (FPS). The word *depth* in the last sentence means the distance between Kinect sensor and the objects in front of it. Kinect sensor consists of an infrared laser projector combined with a vision sensor, which generates image that is based on the distance which is called depth image.

The Kinect sensor can converts a pixel from depth image coordinates to skeleton coordinates which automatically build a human skeleton model in a 3D virtual space and this feature is embedded in the Software Development Kit (SDK) for Kinect sensor. This feature attracted the attention of researchers from HCI because it dramatically simplified the development for innovative HCI application. Although it is mentioned by Obdrzalek (Obdrzalek et al. 2012) that the Kinect sensor has the issue about stability of skeleton information generation, Kinect sensor is worth trialled as an alternative sensor option for pressure sensing.

2.2.5 Summary of Studies on Sensor Technologies

Section 2.2 provided an overview of the pressure sensing technologies that could potentially be utilized by IntelliChair project and the comparison between those technologies is described in Table 2-2 (in the end of this section), furthermore, an alternative vision based sensor technologies is selected other than pressure sensing technologies.

The conductive foam and Force Sensing Resistor sensor are selected and a further experiment is performed to determine which one of them could better fulfil the pressure sensing task for IntelliChair system. Motivated by the result from Brady's group (Brady et al. 2006, Brady, Diamond and Lau 2005), the conductive foam is firstly trialled through experiment in this thesis. Additionally, according to the performance in tacTile mat (Hermann and Koiva 2008) and Mutlu et al's research (Mutlu et al. 2007) the FSR is trialled in sensor selection experiment as well in order to test its reliability and performance. The sensor selection experiment detail is discussed in section 4.1.

The rest of the technology is excluded from the sensor selection list for IntelliChair because of the different limitations. For example, due to the complex amplifier circuits design, EMFi is not included in the experiment. Considering the limitation of the deployment and the output signal feature, the conductive fabric is not included in the experiment as well. Meanwhile, the optical fibre based pressure sensing and the organic transistor system is not included in the experiment because of the relatively high cost, manufacture and deployment difficulty.

For non-intrusive detection purpose, the alternative sensor technology chooses vision based motion sensing instead of inertial sensor based motion sensing. The Kinect sensor is chosen for its simplified development process and its reliability and performance is tested through experiment in section 4.2.

Name of Technology	Authors	Year	Applications	Reliability	Cost
Conductive foam	Brady, et al.	2005	Integrated in cloth for breath rate measurement.	Acceptable reliability.	Very low
EMFi	Lekkala and Paajanen	1999	Integrated in chair for heart rate and emotion detection	Relatively reliable.	Relatively high
Organic Transistor System	Someya et al.	2004	Able to detect both pressure and thermal, for robotic technology	Highly reliable in certain range.	High
Force Sensing Resistor	Interlink Electronics Inc.	2006	Used for sitting posture detection in tacTile mat project and Mutlu's research.	Relatively reliable	Low
Conductive Fabric based pressure sensing	Forlizzi et al.	2005	Used for occupation detection in SenseChair project. Integrated with a belt for heartbeat detection.	Reliable for binary signal, acceptable reliability with circuit support.	Relatively low
Optical Fibre based pressure sensing	Rothmaier, Luong and Clemens	2008	Used for pressure sensing for wearable computing and robotic technology.	Reliable in certain range.	High
	Heo, Kim and Lee	2009			

Table 2-2 Comparison between pressure sensing technologies.

2.3 Discussion

Through the discussion about the pressure sensing based sitting posture recognition systems, it is clear that such system is a part of advancing ambient intelligence and has explored expanding its application to areas such as affective state detection (Mota and Picard 2003), discomfort recognition (Hermann 2005) and low level activity detection (Cheng et al. 2013). The third objective of the IntelliChair project is to build up the correspondence between sitting posture and sitting activities; the experiment for this objective is described in chapter 5 and its experimental design in section 3.5.

The basic structure of a sitting posture recognition system includes the pressure sensing hardware system for data acquisition and certain algorithms for posture recognition on collected data (Tan 1999, Slivovsky and Tan 2000, Tan, Slivovsky and Pentland 2001, Hermann and Koiva 2008, Mutlu et al. 2007, Mota and Picard 2003, Cheng et al. 2013). On the hardware side, researchers are continuing to decrease the number of sensors for the purpose of low-cost without sacrificing the fidelity of the data and this tendency is discussed in section 2.1.4.

In the literature, the minimum sensor number is four (Cheng et al. 2013) sensors from Smart Chair system. Although the Smart Chair system is only sensitive to certain postures, it still shows the possibility that low sensor number could achieve the posture recognition task, as long as the sensors are deployed at correct position.

After the discussion in section 2.1.4, the sensor deployment strategy from tacTiles mat project (Hermann and Koiva 2008) is selected as the hardware structure of IntelliChair project because this sensor placement strategy deploys the sensors on the two surfaces (vertical surface and horizontal surface of the chair) that the human body and the chair are directly contacted, so this sensor placement is able to detect the posture of the different body parts (torso and thigh). Furthermore, according to the discussion in section 2.1.3, this sensor placement deploys the sensors at critical position that is possible to perform accurate posture classification.

In the early stages of posture recognition systems, because of the utilization of a sensor array which has a large number of sensors, those posture recognition system has to perform dimension decrease process in order to avoid the curse of dimension problems (Tan, Slivovsky and Pentland 2001, Mota and Picard 2003, Mutlu et al. 2007). With the decreasing number of sensors, the recognition system is able to perform the classification (Cheng et al. 2013) or clustering (Hermann and Koiva 2008) task directly without dimension decrease process (as discussed in section 2.1.3).

Build training and test data is an important task for posture classification, because it links the physical sensor measurement with sitting postures. Most of sitting posture recognition systems utilise the following procedures, which are also adopted for this thesis. The detailed experimental design is described in section 3.2.4:

- Pre-defined certain sitting postures.
- Label the collected pressure dataset with the pre-defined sitting postures for classification purpose

Subject’s individual characteristics like weight, gender and age is recorded as well, because those information has significant effect on sitting posture.

Once the dataset is collected, it can be used for both training and testing purposes. In the classification evaluation stage, some early research collected an independent dataset for testing (Tan, Slivovsky and Pentland 2001), while the later research utilise the cross-validation method to make best use of the data and simplify the experimental design (Mota and Picard 2003, Mutlu et al.

2007). There are several key points which need to be addressed in the evaluation:

- The number of subjects: this is different in every research, but it shows the coverage of the sitting posture data.
- “Familiar” and “unfamiliar” subjects: this is explained in section 2.1, and it indicates the generalization capability of system to detect the postures with single or multi-subjects.
- Posture sensitivity: this means the sensitivity of a system for certain posture, because of the different sensor placement and classification methods.

Based on the discussion about pressure sensing technology, there are different types of sensor or sensing material with different sensing mechanisms that could be utilized for IntelliChair system. In order to make a decision on sensor choice, there is an experiment for sensor selection in this thesis. The experimental design is in section 3.2.1 while the experiment result is described in section 4.1.

The overview shows that pressure based sensor is the best option for sitting posture recognition, because other motion sensing is focused on the movement measurement like inertial and vision based sensor. But according to the wide usage in pose estimation, vision based sensor, specifically, the Kinect sensor is chosen as an alternative sensor option and it is involved in the IntelliChair project. This thesis also considered the integration of pressure sensing with Kinect sensor because of the purpose to validate the usability and reliability of both sensor systems simultaneously. The integration experimental design is described in section 3.2.2 and the experiment result is described in section 4.2.

Chapter 3 Methodology and Experimental Design

This chapter will describe the experimental methods used in IntelliChair research to address the research questions for IntelliChair project within this thesis.

As discussed in section 1.1, the aim of this thesis is to build up a system that is capable of recognize sitting activity through a non-intrusive and low-cost sensor system. According to the description of the aim, the research objectives in this thesis are:

1. Develop a non-intrusive hardware system that is capable of collecting sensor data at a relatively small financial cost with high accuracy.
2. Detect the sitting posture through the collected data.
3. Establish a correspondence between sitting posture and sitting activities.

Five research questions that are extended from the objectives, which comprise:

1. (Objective 1a): What type of pressure sensor should the system use?
2. (Objective 1b): Are there any alternative sensors to achieve non-intrusive detection other than pressure sensing?

3. (Objective 1c): What is the signal characteristic of the pressure sensing information?
4. (Objective 2): How to classify the data in order to detect the sitting posture?
5. (Objective 3): How to build up the correspondence between the users' sitting posture and the users' activity?

3.1 Approach of the IntelliChair Project

In the literature described in chapter 2, there are already different approaches to the sitting posture detection. The applications covers research area from user's presence detection (Forlizzi et al. 2005) to sitting postures recognition (Slivovsky and Tan 2000, Tan 1999, Tan, Slivovsky and Pentland 2001, Mutlu et al. 2007), Based on the posture recognition, the researchers explored the detection on low level activity (typing on keyboard, nodding head) (Cheng et al. 2013) even the detection of the user's affective states, for example, the user is high interest (exciting), low interest (bored) and taking a break (not playing at all) (Mota and Picard 2003).

Through previous research, there are tendencies which include:

- The simplification of the hardware: With the rapid development of hardware, especially the sensor technology and uniformed middleware platform, researchers are trying to further compact the whole hardware system, in order to make the system easier to be deployed in real situation.

- The interest of high level human activity interpretation: In literature, sitting posture detection is a basic function for such systems, and beyond that, researchers are becoming interested in the more complex human activity status.

As the starting point of early stage research (Tan, Slivovsky et al. 2001) in the literature are focused on biomedical topics, and then it has been expanded and crossed with other areas like novel HCI (Hermann and Koiva 2008) and affective computing (Mota and Picard 2003). The systems (Sensing Chair, tacTile mat, etc.) above all designed as a standalone system to accomplish specific tasks (affective recognition, sitting posture monitoring, etc.).

With the rising concept of Ambient Intelligence (AmI), which is demanding the participation of more individual sensors system, the research of sitting posture detection could be a part of AmI. Imagine a scenario where in a room environment with both IntelliChair and an AmI system integrated, a user has a sitting habit that there will be approximately one hour reading before watching TV when he sits on a couch. When the user is sitting on the IntelliChair mounted couch, the IntelliChair system could determine whether this user is reading based on the detected sitting postures. Based on this user's sitting habit, the system will send request to the AmI system to cooperate when one hour is reached, and ask the AmI system to switch on the TV. This scenario shows IntelliChair is able to assist the users to shift their activities naturally based on their individual sitting habit.

IntelliChair is proposed as an approach, which combines sitting posture classification and sitting activity recognition. The goal of the system is to track the naturally occurring sitting postures of a user through the use of non-intrusive, low cost, chair surface-mounted sensors and establish the correspondences between sitting postures and sitting activities. Potential activities include relaxing, watching TV, play games and the range can be extended because the focus of this thesis is to compare the predicted activities to the natural occurred activities recorded by the subjects. A behaviour pattern is a sequence of ordered activities that frequently occur together. Those behaviour patterns can be used to provide personalized service based on individual sitting habit. The following section introduces the detail of the IntelliChair system requirements along with the IntelliChair hardware system design.

3.1.1 IntelliChair System Development Requirement

According to the description above, the architecture of the IntelliChair is proposed as shown in Figure 3-1 which includes the following components:

- Local system: Combination of sensors, related circuit and DAQ unit with a compact computation system that is capable for posture classification process (Left part of Figure 3-1). This allows IntelliChair could provide sitting posture information independently.
- Remote system: Receive the posture classification result as input, estimate the posture and activity relation and recognise the activity. This allows the computation consuming activity recognition process stay in the cloud with more computation resource.

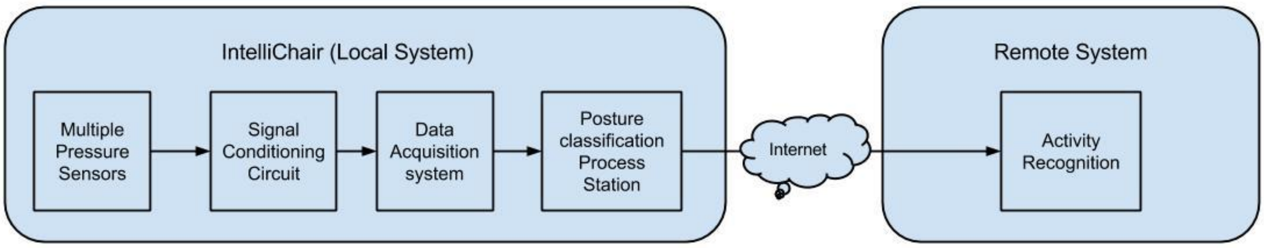


Figure 3-1 The proposed system architecture for IntelliChair system.

The local system consists of major hardware of the IntelliChair which includes four parts: Pressure sensor, Signal Conditioning Circuit, Data Acquisition (DAQ) system and computation unit for posture classification.

Pressure Sensor is the core part for pressure sensing, and the options is listed in section 2.3. The requirement for the sensing unit includes:

1. Capable to response to pressure change within seconds, because IntelliChair is required to provide near real time classification result.
2. Relative stable pressure measurement: According to Brady's research (Brady, Diamond and Lau 2005) and FSR integration guide (Interlink Electronics 2006), both sensors in the option list has the measurement drifting problem, so certain measurement deviation in the experiment is acceptable, but the difference should not be too great.
3. Measurement repeatability: Because the sensor is deployed at critical position of the chair, so the change of posture could leads to major pressure measurement difference of a sensor at a particular position (e.g. the sitting straight posture to crossing leg posture). So repeatability is not a major issue but still a factor for sensor selection.

Signal Conditioning Circuit is co-ordinate the signal from sensor, transform the pressure measurement into digital signal that could be collected by DAQ system. Furthermore, the circuit is required to modify the relation between digital signal and pressure as linear as possible. DAQ system is the direct receiver of the digital signal, it is required that the DAQ system for IntelliChair must have 8 input ends because there are 8 sensors according to the sensor placement strategy determined in section 2.3. DAQ system should have a minimum overall sampling rate of 8 Hz (this ensures that for each sensor, it has at least 1 Hz sampling rate). The Signal Conditioning Circuit and DAQ combination is the middleware of the IntelliChair and it is required that it is capable to be connect with other sensor system because as discussed in the section 2.2.4, there are alternative sensor like visual based sensor, and it is part of this research that two different type of sensor could integrate together (software integration, not hardware integration).

Posture Classification Process Station is a component that is required to running classification algorithms, so it should had processor, memory, RAM for storage and an Operating System for algorithm programming.

The software requirement for this research includes the hardware interfacing (sensor data retrieval), database operating (data storage), GUI programming (for data visualisation and user interface for experiment participants) and algorithm programming for classification (sitting posture) and pattern recognition (sitting activity). Several software and programming language are used in this research, but Python (Python Software Foundation 2015) is the major programming language used throughout the research.

3.1.2 Work on Hardware Middleware

Following the development requirement in the previous section, the hardware (middleware) of IntelliChair is discussed in this section. The signal conditioning circuit for pressure sensor and different correlated data acquisition systems is introduced and discussed in order to determine the suitable middleware for IntelliChair.

There are three parts to the process that passes information from the physical phenomenon to a computer system. These are transduction, signal conditioning, and data acquisition (Putnam and Knapp 1996) as showed in Figure 3-2.

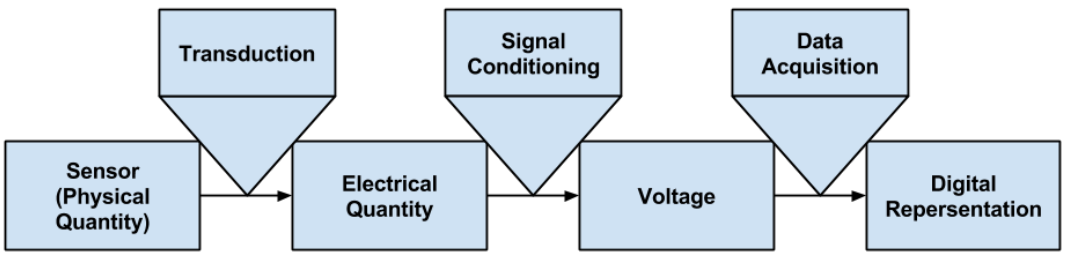


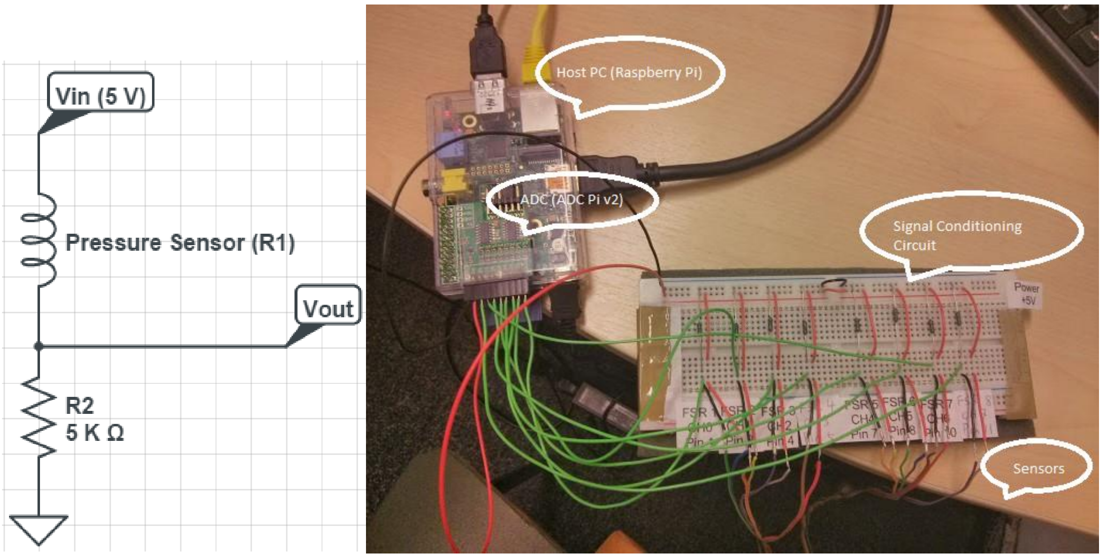
Figure 3-2 The three parts of data collection from sensor to computer system (Putnam and Knapp 1996).

The sensor or transducer aims to provide the transduction from physical quantity to electrical quantity (usually voltages). After the transduction, the electronic quantified information is changed to an appropriate form (voltage, resistance, etc.) and this is the signal conditioning. The purpose of the signal conditioning circuit is to remove unwanted signals, modify the sensor's spectrum (for example, shape the correspondence between physical quantity and electrical quantity from logarithm to near linear relation), and map the sensor output to the data acquisition input. The last part is the data acquisitions, which records the continuous signal that is measured by sensor and modified by signal conditioning circuit in a form in order to be analysed by other software.

Since the technology of pressure sensor and transducer is discussed in section 2.2, this section introduces the signal conditioning circuits for selected pressure sensors, and the correlated data acquisition systems that are used in this research.

3.1.2.1 Signal Conditioning Circuit

According to the integration guide of Force Sensing Resistor (FSR) (Interlink Electronics 2006), a signal conditioning circuit for translating the resistance change into voltage change which is shown in figure 3-3.



(a) (b)

Figure 3-3 The Voltage Divider circuit for a single Pressure Sensors (a) and the integration with DAQ system (b).

The equation for the voltage calculation is:

$$Vout = \left(\frac{R_2}{R_1 + R_2}\right) * Vin$$

In this equation, the $Vout$ is the voltage measurement that will be collected by Analog-to-Digital Converter (ADC). R_2 is the voltage divider resistor, and its value is 5k Ohm constantly and Vin is the reference voltage, and is constantly 5 V. As the previous description of pressure sensor, when force is increased, the resistance R_1 will decrease. Refer to this equation, it means the R_1 will decrease when force is increase. Since R_2 and Vin are constant, so $Vout$ will increase as well as the applied force. So the changing of force is represented by $Vout$ in the record data.

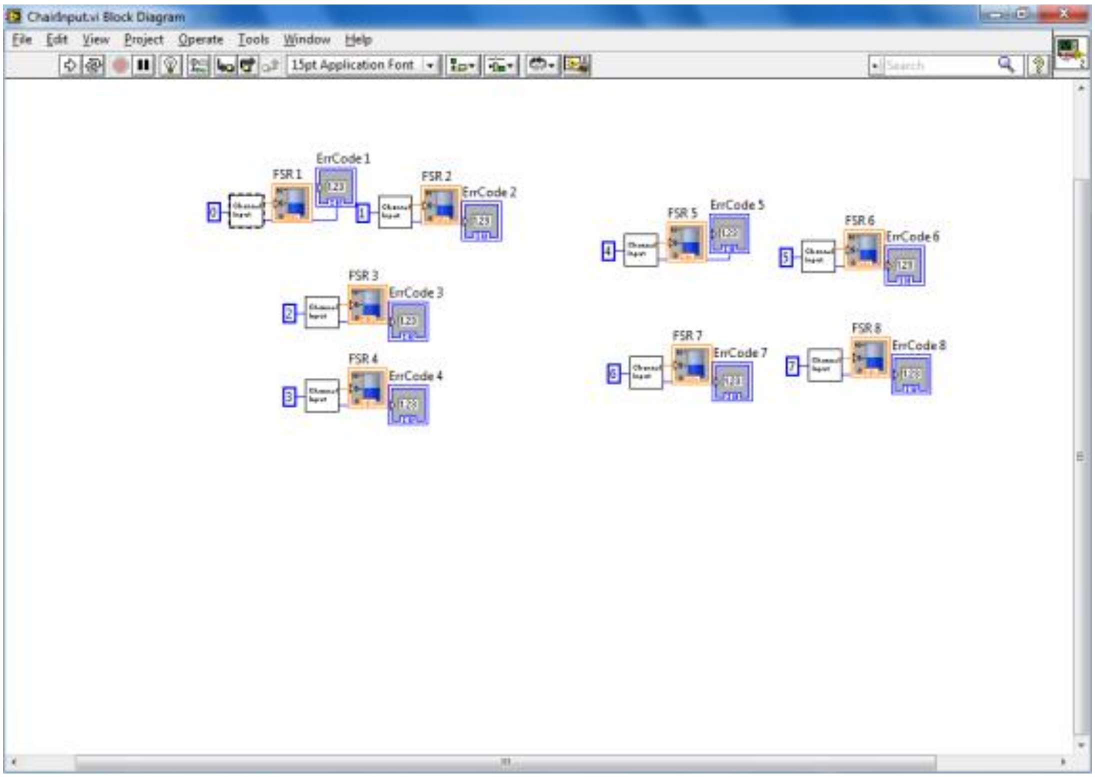
3.1.2.2 Hardware and Correlate Software Selection

In this section, three different hardware platforms with their correlate software that are used in IntelliChair development are introduced, they are:

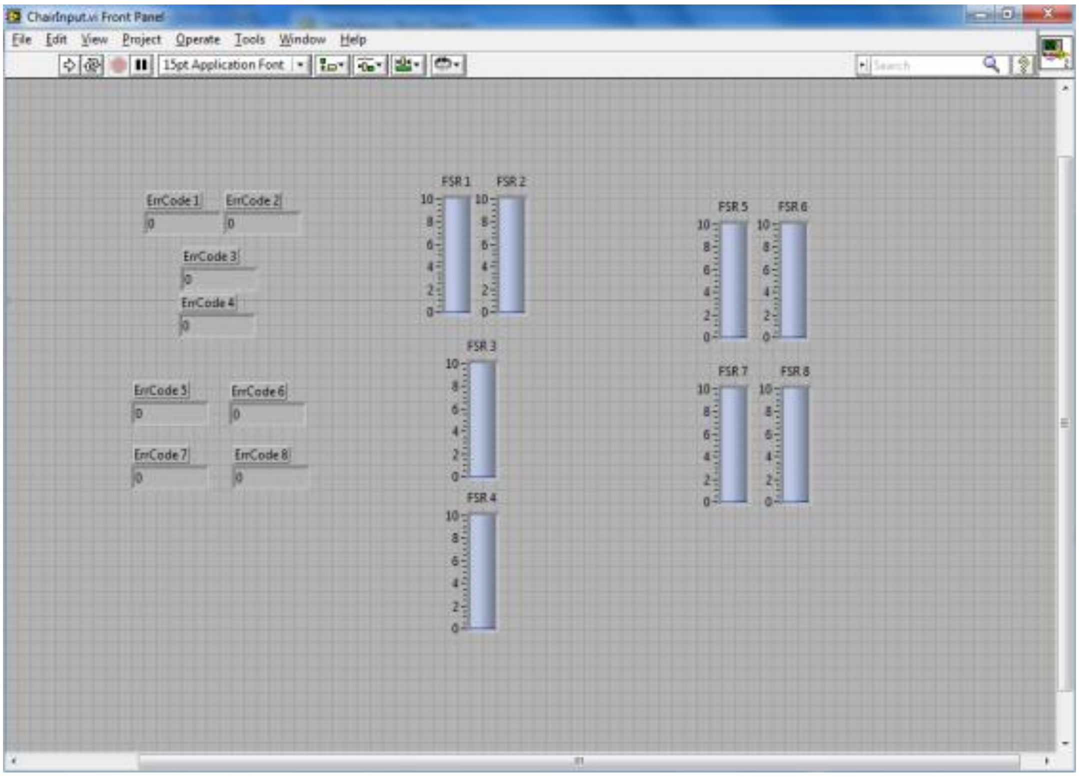
- USB 1208-LS and LabVIEW
- Arduino and Python
- Raspberry Pi and Python

USB1208-LS is a data acquisition device which is produced by Measurement Computing TM (Measurement Computing Corporation 2015).USB1208-LS has 8 analog input channels and maximum 1 MS/s sampling capability which ensure the sensitivity of signal sampling. This device is supported under Microsoft Windows operating system and relies upon the device driver from Measurement Computing Corporation (MCC). With MCC Universal Library installed, this device could further compatible with LabVIEW Integrated Development Environment (IDE).

LabVIEW is visualised system design software produced by National Instruments Corporation (National Instruments Corporation 2015). LabVIEW IDE provides comprehensive tools and support for building measurement or control application development. Figure 3-4 shows the USA1208-LS device and the correlated IntelliChair application in LabVIEW IDE.



(a)



(b)

Figure 3-4 The IntelliChair application in LabVIEW IDE.
(a) The block diagram of IntelliChair system application (How data flows). (b) The front panel of IntelliChair system application (The display of the sensor measurements).

The combination of USB1208-LS and LabVIEW provides a visual application development process with easy device reconfiguration. But its drawback also comes from LabVIEW platform because it is difficult to interact with other programing languages to extend its function because it relies on payment-required add-ons, which increase the development cost.

Arduino (Arduino 2012) is an open-source hardware platform board with integrated microcontroller. The type of Arduino board that is used in this thesis is Arduino Mega. The program that runs on Arduino is called sketch, a Java style syntax language, easy to learn with full document support from Arduino community. However, it needs a host PC to link with the Arduino board in order to communicate with Arduino board (receive and send data).

Python (Python Software Foundation 2015) is chosen as the data acquisition programming language on the host PC because it is capable of hardware access, its cross-platform feature and it has support for scientific and numeric computing. Furthermore, it has a large number of libraries and packages and an active developer community. A python package named pySerial (Liechti 2013) is utilised for the communication between Arduino and host PC program.

The utilisation of Arduino and Python aims to build a system that could visualise pressure information. The system is able to merge the data from Kinect sensor system to further visualize the human skeleton information along with the posture pressure information. The detail about the data visualization experiment is described in section 4.2.

Although Arduino is capable of some logical computation, its computation capability is at microcontroller level rather than a processor level. Furthermore, the host PC dependence feature makes the Arduino version of IntelliChair system difficult to be deployed in real world experiment.

In order to further compact the system, the system is shifted to Raspberry Pi platform. The Raspberry Pi (Raspberry Pi Foundation 2013) is a low-cost, credit card sized single board computer, with Linux-based Raspbian operating system. Raspberry Pi can perform the task of a host PC; with the extension board “ADC Pi V2” (AB Electronics UK 2013) , Raspberry Pi is able to acquire the information from IntelliChair sensors and can undertake the posture recognition task locally.

Because of the cross-platform feature of Python, it can still support the data acquisition on the Raspberry Pi. Although heavy computation consumption tasks like sitting activity modelling and recognition is running on remote server, the Raspberry Pi system is able to tackle the tasks like posture classification based on the trained classifier. The detail of the IntelliChair modelling and recognition architecture is shown in Figure 5-1 and Figure 6-1.

3.1.2.3 Summary of Hardware Middleware

Section 3.1.2 introduced the signal conditioning circuit for pressure sensor and described the data acquisition systems that are utilised in this thesis. Throughout the research, if the data acquisition system change is needed, only the electrodes are required to re-connect the correct analog input pin on another Data Acquisition component (e.g. from Arduino to ADC Pi V2).

IntelliChair employed the USB1208-LS and LabVIEW combination in the early development stage in order to assess the feasibility of the signal conditioning circuit and data acquisition. The Arduino version IntelliChair system aims to provide the interface for integration with Kinect sensor for real-time data visualisation.

The Raspberry Pi aims to further compact the system and extend the further usage in real experiment deployment. The Raspberry Pi version of IntelliChair integrates the process station (Raspberry Pi) along with the DAQ (ADC Pi V2) system; this integration not only simplified the software development process (the sitting posture classification program is possible to integrate with sensor data collection) but also make IntelliChair closer to real life usage.

3.2 Research Questions and Challenges

In order to achieve the aim of the IntelliChair project which is discussed in section 1.1, there are five research questions:

1. (Objective 1a): What type of pressure sensor should the system use?
2. (Objective 1b): Are there any alternative sensors to achieve non-intrusive detection other than pressure sensing?
3. (Objective 1c): What is the signal characteristic of the pressure sensing information?
4. (Objective 2): How to classify the data in order to detect the sitting posture?
5. (Objective 3): How to build up the correspondence between the users' sitting posture and the users' activity?

3.2.1 Sensor Selection

The selection of pressure sensors is discussed in the literature in section 2.2.5. Although the posture sensing mechanic is chosen to be the force or pressure based, there are many ways to achieve the detection. Some project developed force sensing matrix mat (Tan, Slivovsky and Pentland 2001, Mota and Picard 2003, Ciampone 2012) as described in section 2.1.1, while others deployed individual sensor at a critical position (Hermann and Koiva 2008, Mutlu et al. 2007, Forlizzi et al. 2005, Cheng et al. 2013) as described in section 2.1.2 and 2.1.3. The former could provide an accurate pressure distribution image, while the advantage of the latter is the systems simplicity and its low cost on hardware. Following the research result of Mutlu (Mutlu et al. 2007) from Carnegie Mellon University, it is possible to replace the Tekscan BPMS (total 4032 sensors) into a set of 19 sensors, with limited sacrifice of the sitting posture classification accuracy (see section 2.1.2). The later researchers like Hermann (Hermann and Koiva 2008) and projects such as Smart Chair (Cheng et al. 2013) also proved that correct sensor placement strategy is capable of sitting posture classification.

Although the critical position sensor placement strategy is proved to be capable of sitting posture classification, the classification performance is still affected by the sensor numbers. For example, according to Table 2-1 in section 2.1.4, the classification result of Mutlu's group (Mutlu et al. 2007) is claimed to be 78% with 10 postures among 20 subjects (19 sensors in the system), while the Smart Chair project (Cheng et al. 2013) claims the classification result is 82.6% with 7 postures among 5 subjects (four sensors in the system). It seems the Smart Chair has better result, but if looking into the Smart Chair's classification result detail which shows in Figure 2-8 in section 2.1.3, it can be found that Smart Chair is only sensitive to certain classes of sitting posture like sit straight, lean left, lean right and lean forward.

On the opposite side, Mutlu’s 78% result is more promising because they use two “unfamiliar” subjects (among 20 subjects, the rest 18 subjects’ data for training) to test the classification performance.

Also, the comparative result might be relevant with the selection of the sensor product as well. The Smart Chair project employs the handcrafted conductive foam based pressure sensor as showed in Figure 2-8 in section 2.1.3. The mechanism and performance of such sensor material is described in the Brady’s research (Brady, Diamond and Lau 2005). The conductive foam that is used in our research might not be the same as the material used in Smart Chair project and Brady’s research, but their mechanism is the same (when force is applied, the resistance of the material decreases along with the increasing force). On the other hand, Mutlu’s research (Mutlu et al. 2007), as well as Hermann’s research (Hermann and Koiva 2008), used a commercialised force sensing product named Force Sensing Resistor (FSR). In order to make a choice, it needs to decided what pressure sensor should the system use?

There are many kinds of pressure or force sensors exist in the market, from strain gauge to vacuum gauge. Although those types of sensor are small, inexpensive and have relatively reliable accuracy, they are designed for structural measurement in the field like building construction or vacuum pressure measurement inside containers.

The situation is different while dealing with human aspect. Especially in the HCI area, the pressure or force sensor is preferred to be more flexible on the structure in order to tackle the flexible tactile surface, and the demand on the sensing range is much smaller compare with conventional type of sensors. So instead of robust circuit and board design, novel pressure or force sensors are more focused on the sensor adaptability in complex human tactile sensing situation. For example, Brady's group (Brady et al. 2006, Brady, Diamond and Lau 2005) developed conductive polymer based pressure sensor which is capable to be integrated into clothing which is discussed in section 2.2.1.

Also, in the literature, there are other different approaches of novel sensor design which are capable of force or pressure sensing as discussed in section 2.2.1 to section 2.2.3. From conductive polymer or conductive fabric based sensor to optical fabric matrix based artificial skin, and there are also many commercialised product like pressure sensing mat series product from Tekscan, Force Sensing Resistor from Interlink Electronics. Each type of sensor has their focus and field of usage. Which one is the best choice for IntelliChair project?

Consider the aim of IntelliChair project mentioned in section 1.1 and the discussion about the previous chair systems, the sensor should include features like:

- Inexpensive hardware cost.
- Relatively stable measurement in human tactile situations.

Under these circumstances, the scope for the sensor selection is narrowed. According to discussion in section 2.2.5 and Table 2-2, the BPMS from Tekscan is not ideal for its high cost, although it has the best performance and the most complete data analysis software support. The optical fibre based sensor (Heo, Kim and Lee 2009) and organic transistor system (Someya et al. 2004) are also removed from the list because their manufacture difficulty, there are no commercialised product for this type of sensor and their manufacture requires the Silicone process facility which the university could not provide. Conductive Fabric is not included because of its binary output signal feature, and EMFi is not included because of its cost and complex amplifier circuit design.

This leaves the Conductive foam material based sensor and the FSR products. In order to evaluate the performance of two different sensors, an experiment is carried out. The experimental design is described in section 3.3.1 and the experiment result is described in section 4.1. The experiment evaluates the pressure sensing performance of the conductive foam material as the sensing unit as a sensor and it also evaluate the pressure sensing performance of the Force Sensing Resistor.

3.2.2 Alternative Sensor Integration and Visualization

The second question addresses the situation when the sensor selection is extended to a wider range of sensor types other than pressure sensing which is discussed in section 2.2.4. Pressure or force sensing is not the only way to achieve the non-intrusive sensing; it also includes vision based sensing as well. Especially the rising attention on new uniformed vision based sensor, the Microsoft Kinect sensor, which has the advantage of non-intrusive user experience.

The attempt to merge different type of sensor system for seated posture detection already existed before Kinect sensor got attention from researchers. In Ciampone's research (Ciampone 2012), vision based posture classification plays an important role. The vision based sensing method in this research is to use a camera which positioned to the side of the subject and some markers attached on the joint of subject's body to acquire the visual information, and using image processing technique to build the body parts (head, neck, leg, foot etc.). By calculating the angle between each two body parts (head and neck, leg and foot, etc.), the subject's sitting posture could be determined.

Ciampone's system uses the markers to identify the joint points of the human body as an anchor point for the system to build the body segment. From this perspective, this system is not a non-intrusive system. As introduced in section 2.2.4, the depth information based vision sensor, Kinect sensor does not require any marker as an anchor point for body segment modelling, which better serves the objective of non-intrusive sensing.

According to the discussion above, there is an experiment in this thesis with two objectives: Evaluate the usability of the Kinect sensor for sitting posture detection. Then, explore the possibility to integrate the Kinect sensor data with pressure sensor data by visualize them together in real time. In order to tackle those requirements, a pressure sensor data and vision based data integrated system is developed in this thesis and this system is able to visualize both two types of data. The data visualization is able to validate the reliability of vision based sensing and pressure sensing, meanwhile, by integrating both types of data, the reliability can be compared at same time. The detail about the experimental design is described in section 3.3.2 and the experiment result is described in section 4.2.

3.2.3 Posture Signal Characteristic

In the literature, there are many discussions and results about posture classification from section 2.1.1 to section 2.1.4, but there is a lack of detailed discussion and explanation that covers the sampling frequency of the hardware system, and why to choose a specific frequency rate. And this is the third research question for the IntelliChair project. By considering the posture information as a signal, and apply the signal processing method with the experiment data to discover the best sampling frequency range for the sitting posture collection.

Since the purpose of this experiment is to explore the frequency characteristic of the pressure information generated from human sitting posture, the method, frequency-domain analysis is utilised. Frequency-domain analysis is a tool that is widely used in signal process applications like image processing, communication, etc. The function of this analysis is to convert the signal from the time-domain to the frequency-domain. The difference between time-domain analysis and frequency-domain analysis is that the time-domain analysis indicates the information about a signal changes over time, but the frequency-domain analysis shows the distribution of the signal's energy over a range of frequencies. The details about the time-domain and the frequency-domain is shown in Figure 4-12 in section 4.3

The mathematical operation to achieve the conversion between time-domain and the frequency domain is a Fourier transform. It decomposes the frequency information of a signal into a sum of a number of sine wave frequency components. The set of frequency components represents the signal's frequency information, and among those components, there are some components which show significant importance for frequency representation. Those important frequency components are the key to determining the frequency features and furthermore, determining the sampling frequency for a signal.

In the experiment, the pressure information which is generated by human sitting posture is considered as a signal, and the data collected from the IntelliChair system is time-domain based. By using the Fourier transform method, the raw data is converted from the time-domain signal data into frequency-domain and the frequency domain information will be visualised. The frequency characteristic of a human sitting posture signal can be discovered from those figures and hereby, the sampling frequency for the sitting posture could be determined. The detail of the experimental design is described in section 3.3.3 and the experiment result is described in section 4.3.

3.2.4 Posture Recognition

Classification is considered to be the most widely used method in the sitting posture analysis area (Tan 1999, Tan, Slivovsky and Pentland 2001, Mota and Picard 2003, Cheng et al. 2013, Mutlu et al. 2007, Forlizzi et al. 2005). In the literature, only Hermann’s research (Hermann and Koiva 2008) use clustering as the analysis method, because they do not need to know the specific posture, instead, their point is monitoring the change of a posture.

Based on machine learning and data mining principles, “classification is about determining the class of a given data record by applying a classification model. Such a model is developed from a set of data records know as examples, and each example consists of a list descriptive features and a class label that is already known” (Du 2010). The word Example is mainly used in data mining area, while in the machine learning area, it sometimes be called as training data. In the rest of the thesis, the phrase “training data” will be used.

The process to achieve the classification was described in section 2.1; it includes two stages:

- Build the Classification Model: A classification model can be built through model-building methods using the training dataset.
- Classification model based decision making: After the classification model is built, the model can then make decisions that will classify a new example (no matter unseen or not) into pre-defined classes.

In order to build a classifier that is capable of detecting different known postures, the first thing to do is to collect a training dataset. This training dataset should include both example data and a corresponding label dataset that indicates which of the class the example data is allocated. After being trained by using a subset of the training dataset, the classifier can be used to predict new incoming examples, and determine which class the new example belongs to.

Not all the training data will be used for training because some data is used for classification evaluation. In practical terms, the amount of data for training and testing is limited, and researchers found the best way to make full use of the data is to set part of the training data for training, and the rest for validation. This method is called Cross-Validation (Witten and Frank 2005).

It is common to use “threefold cross-validation” (Witten and Frank 2005) where one-third of the data is held for evaluation, and the other two-thirds for training. But through many research results, the best solution is tenfold instead of threefold. It means the whole training dataset will be randomly divided into ten parts and each part is used as validation while remaining nine-tenths is used for training. With ten evaluation results, an average evaluation can be estimated, and the result is proven to be more reliable compared with three-fold (Witten and Frank 2005).

Since classification is a developing research field in machine learning and data mining, there are many different kinds of classification algorithm or modelling methods to deal with different types of data. In this thesis, the form of the sample data is different from the projects which have high dimensional data in the literature (Tan, Slivovsky and Pentland 2001, Mota and Picard 2003, Mutlu et al. 2007). It requires that the posture classification task is performed locally, so the classification algorithm Support Vector Machine (SVM) (Witten and Frank 2005) is utilized and its classification performance is evaluated. The reason for this choice is because of the decision function of

SVM relies on the Support Vectors instead of the number of training samples, which makes the SVM based classifier is capable to perform classification task on limited computation capability device such as Raspberry Pi. The details about SVM is described in section 4.4.3 and section 4.4.4.

In the literature, how to deal with data from multiple subjects is considered as a part of sitting posture research (Slivovsky and Tan 2000, Tan, Slivovsky and Pentland 2001, Mutlu et al. 2007). This is for the evaluation of the classification performance when system deals with new subjects. So in this thesis, the data will be collected from multiple subjects. The experimental procedure to collect the training dataset and the result of the classification performance evaluation is described in section 4.4.

3.2.5 Activity Modelling and Recognition

In the literature, there are different focuses about the extensive research on human sitting postures such as affective state detection (Mota and Picard 2003), emotion detection (Anttonen and Surakka 2005), low level activity detection (Cheng et al. 2013),etc. In this thesis, the focus is the activities when human is sitting on IntelliChair. The “activity” means the current status (Working with PC, reading, eating, relax, sleeping, etc.) of a human subject which can be referred to the watching TV scenario in section 3.1. According to the literature, some researchers try to interpret the activity by correlating it with pressure information (Cheng, Zhou et al. 2013), and in some literature, researchers focus on the subject's physical condition measurement like heart rate (Anttonen and Surakka 2005) rather than the long term human activity states.

Inspired by the behaviour detection in Ambient Intelligence (Monekosso and Remagnino 2009), the activity modelling could also be done by correlating the posture and activity. Because in Monekosso and Remagnino's research, it is assumed that the human indoor behaviour is formed by a set of time-ordered activities (sleeping, cooking, eating, etc.). The essence of their hypothesis is to interpret a model according to time-ordered sequential data. Refer to this thesis; it can be assumed that the sitting activity can be correlated with a set of time-ordered sitting posture sequence.

The modelling method for this sitting activity and sitting posture correspondence is the Hidden Markov Model (HMM) which is also used by Monekosso and Remagnino (Monekossoand Remagnino 2009). The detail about this model is described in section 5.1. The way to build the correspondence between activity and posture is to recognize the activity based on the output of the classification stage, which specifically, is a sequence of postures.

In order to build up an activity recognition component, there are three steps. Firstly, a set of independent HMMs are trained, and each HMM corresponds to a specific activity (relaxing, working with PC, etc.). Secondly, each trained HMM computes the probability when a posture sequence is input. Thirdly, the output activity is determined by the highest probability of a HMM set. It means that according to the input posture sequence, the activity recognition component categorizes this posture sequence as belonging to a specific activity. Through this recognition process, the posture sequence is abstracted into an activity sequence, and the relationship between posture and activity is established.

3.3 Experimental Design

3.3.1 The Experimental Design for Sensor Selection

The experiment's purpose is to answer research question 1 (objective 1a): what type of pressure sensor should the system use? This experiment therefore evaluates the usability of a pressure sensing material (conductive foam) and the performance of force sensing resistor sensor (FSR) which were introduced in section 3.2.1. The comparative results of the evaluation are expected to provide the support for decision making of the pressure or force sensor selection. The conductive foam is from Maplin Electronics. Inc and the FSR is a product from Interlink Electronics .Inc (product number FSR-406).

The experiment is divided into two parts; the first part is for the usability of the conductive foam as pressure sensing material, the second part is to evaluate the performance of the FSR sensor. Two major focuses of this experiment are:

- Is the measurement stable when a specific weight is applied over time?
- Could the measurement from the sensor be maintained when a specific weight is applied to the sensor again after weights changes?

The first focus is because in some situations (watching TV, play games, etc), the human sitting posture will stay still for a period of time, and this test is trying to evaluate the performance when the foam and the sensor is dealing with certain weights in a period of time. The second focus is to examine the stability of the sensor measurement with the specific weight, after a process that different weights are applied to the sensor (for example, sensor measurement is A when 1KG weight is applied, after a process of weight changes from 0.5 KG to 2 KG, then 1KG weight is applied to on the sensor again, could the sensor measurement still maintain at A?).

Based on the experiment experience from Brady's research (Brady et al. 2006, Brady, Diamond and Lau 2005), some preliminary experiments were carried out to give a general impression of both conductive foam and FSR sensor. The experience from the preliminary experiments confirms that both conductive foam and FSR sensor suffer from has the measurement drifting problems reported by Brady's research (Brady, Diamond and Lau 2005, Brady et al. 2006) and the FSR sensor datasheet from Interlink Electronics.Inc (Interlink Electronics 2006). This phenomenon is called resistance drift, it means the resistance will change (normally decrease) with time under a constant (static) load. It happens on both conductive foam and the FSR sensor because they both rely on the resistance changing when force is applied to the surface of the pressure sensing material. When the material is compressed, the conductive condition is changed, which makes the resistance change as well, no matter whether the force applied is static or not.

So as long as the force is applied on sensor, the sensing material is always under compressed condition, which leads to constant resistance change.

The performance of FSR is much better than the conductive foam, because the FSR sensor's measurement is quickly stabilized at some point around 10 seconds. Drifting then occurs but the measurement decreases very gently and slow; according to the datasheet of Interlink Electronics, the resistance decreases is less than 5% in ten days' time (Interlink Electronics 2006). But the situation is different in conductive foam preliminary experiment while the resistance measurement decreases constantly after around 15 to 20 seconds there will a short period of time that the measurement decrease will slow down, after which the measurement decreases quickly again. Another phenomenon is that if both the sensor and the foam is being compressed by a higher weight, then after the release to a lower weight, the resistance measurement could responded to the weight change with increasing resistance measurement, but the measurement is not equivalent to the measurement before compressed, usually the former measurement is higher than the latter one.

With the results from the preliminary experiment, the experiment procedure is determined. The experiment procedure is to increase the weight applied on the foam or FSR sensor by 1KG steps (starting at no weight applied) to a maximum of 8KG and then to reduce in 1KG steps back. The reason for this weight range (1 KG to 8 KG) is because it covers the force sensitive range of FSR sensors which is listed in its data sheet (Interlink Electronics 2006). The force sensitive range of the conductive foam is unknown, but because the sensing mechanic is similar for both foam and FSR sensor, this range should cover the foam's capability force sensitive range as well. The whole procedure combines the resistance measurement under constant weight and the resistance measurement with different weight.

In order to further discover the characteristics of FSR sensor, there will be an additional experiment for FSR, same procedure but the range of the weight increase and decrease is narrowed as well as the weight change for each step. The detail of the experiment is described in section 4.1, and the result will be explained as well.

3.3.2 The Experimental Design for Alternative Sensor Integration and Visualization

This experiment's purpose is to answer the research question 2 (objective 1b): Are there any alternative sensors to achieve non-intrusive detection other than pressure sensing? Refer to the objectives in section 3.2.2, the expectation for this experiment is to validate the usage and the reliability of Kinect sensor by develop a dynamic, real-time data visualisation system that could display both skeleton data and pressure data when user is sitting on IntelliChair. This system is designed that including three components:

- Kinect sensor component.
- Pressure sensing component.
- Data visualization component.

The Kinect sensor used in this experiment is the first version of the Kinect with XBOX, and the Kinect SDK version is 1.0. One requirement of this SDK is that it must be installed on Microsoft Windows Operating System version 7 or higher. Considering this requirement, the pressure sensing component should also deliver the pressure data to Windows OS installed process station, and the data visualization should be developed in Windows. The proposed system architecture is in Figure 3-5.

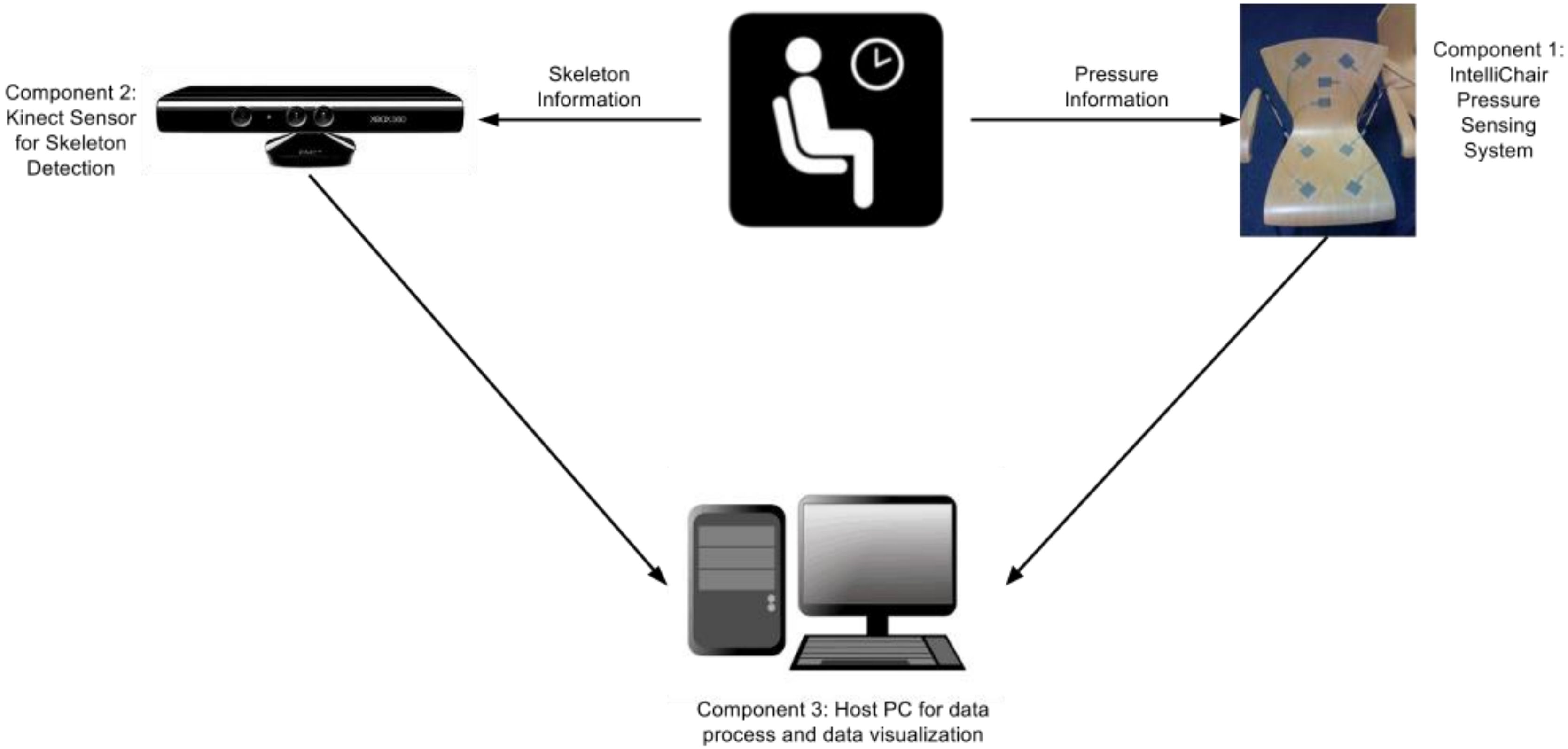


Figure 3-5 The architecture of the sensor integration system.

In order to correlate with the Kinect sensor component, the Arduino mega board is utilized to connect with the middleware of the IntelliChair pressure sensor component. As discussed in section 3.1.4, the Arduino mega board collects the analog data from eight pressure sensors (sensor placement strategy is discussed in section 2.1.4) and delivery the data to the host PC. The Kinect sensor is also connected to the host PC and frequently posts the skeleton information at 30 Frame per-Seconds (FPS).

The data visualization component is developed based on Python in order to unify the software development. There are three existing Python packages that deal with different tasks:

- pySerial (Liechti 2013): This package is used to handle the data that is delivered by the Arduino Board.
- PyKinect (Microsoft 2013): This package is a Python binding of the Kinect sensor SDK, which could access most of the Kinect sensor functions. It will acquire the skeleton information from Kinect sensor and deliver to Host PC.
- VPython (National Science Foundation 2012): This package is capable of building a 3D visual space, with easy and flexible object development within the space.

Using the three packages, the system is able to display the real-time pressure data from Arduino and skeleton data feed from Kinect sensor in a 3Dimensional visual space. The experiment result is described in section 4.2.

3.3.3 The Experimental design for Posture Signal Characteristic Analysis

This experiment's purpose is to answer research question 3 (objective 1c): What is the signal characteristic of the pressure sensing information? The objective of this experiment is to explore the characteristic of the posture signal and find a suitable sampling rate for IntelliChair system to collect the pressure information. The reason that only pressure information is used is because only pressure sensing is used in final setup (see section 4.2).

Through the description in section 3.2.3, the method to uncover the signal feature is to convert the signal from time-domain to frequency domain, especially the plot for the frequency domain, because this plot highlights the significant frequency components which show as peaks among the curve. The maximum frequency is determined by the last significant peak, and this is the key feature looked for.

According to the Sampling Theorem (Weisstein 1999), in order to collect lossless information for signal re-construction from a signal source, the sampling frequency of a collecting system should be larger than twice of the maximum frequency responses, and twice of the maximum frequency responses from the signal source is called Nyquist Rate (Weisstein 1999). The higher the sampling frequency, the more additional information will gain. The Nyquist Rate is the suitable sampling rate this experiment is looking for.

In addition, considering there are different contact patterns (e.g. rapid leg shake like stroke, frequently contact the chair, or sit still for a long time, only micro muscle changes.) between human and chair, their signal characteristics (maximum frequency and Nyquist Rate) might be different as well.

Based on the discussion above, the experiment procedure is determined:

1. Human subject (author of this thesis) is asked to sit on the IntelliChair and perform different human-chair contact patterns (see section 4.3) for 10 seconds each.
2. The data, which is the frequency changed pressure, will be recorded in time order.
3. After the data collection, the dataset is converted into an array by using the

Python Numpy package, and because the Fourier transform function is also integrated in the package, by invoking fft() function, the whole array can be converted into frequency domain information.

4. The Frequency domain plot is draw in Matlab, and used to find the peaks to determine the maximum frequency.
5. Determine the Nyquist Rate for each human-chair contact pattern by using the maximum frequency from last procedure.

Along with the development of IntelliChair system, there are two stages within the experiment. The first frequency experiment is based on the initial IntelliChair system with relatively lower maximum sampling rate and its purpose is a preliminary exploration of the human sitting posture pressure signal. After the modification and improvement of the IntelliChair system, especially the higher sampling rate, the second frequency experiment aims to achieve the maximum sampling rate in order to learn more about the pressure signal detail. In second stage experiment, different types of signal are implemented, which covers high and low frequency contact and different body parts between human body and IntelliChair sensor system. The detail of the experiment is discussed in section 4.3.

3.3.4 The Experimental design for Sitting Posture Detection

This experiment's purpose is to answer research question 4 (objective 2): How to classify the data in order to detect the sitting posture? The objective of this experiment is to collect the pressure information based dataset with specific posture labels, in order to build a dataset to train the IntelliChair system that enables the system to classify different sitting postures. Table 3-1 shows the pre-defined posture names and their labels. There are total 16 pre-defined sitting postures in this thesis, the first four postures (1 to 4) are upper body part (spine, torso) postures, and the rest are lower body part (hips, legs) postures (5-16).

Posture Group	**Postures**	**Label**
Spine Postures	Body Lean Right	1
	Body Lean Left	2
	Body Leaning Back	3
	Body Crouch	4
Leg Postures	Sitting on Edge	5
	Crossing Right Leg on Left Leg	6
	Crossing Left Leg on Right Leg	7
	Sitting Forward	8
	Sitting Forward Left	9
	Sitting Forward Right	10
	Sitting Upright	11
	Sitting Upright Left	12
	Sitting Upright Right	13
	Leaning Back	14
	Leaning Back Left	15
	Leaning Back Right	16

Table 3-1 The pre-defined postures and their corresponding labels.

The number of postures is summarised from the literature, especially from Tan's research (Tan, Slivovsky and Pentland 2001, Tan 1999) in section 2.1.1 which defined 10 postures. The difference between this thesis and Tan's research is that in Tan's research the human body is considered as a whole, so the posture they defined consists of both body and leg parts. But in this thesis, the sitting posture is divided into two parts: body postures and leg postures. The consideration for the separation of the body segments is due to two reasons:

- The sitting posture is a combination of body and leg segment, for example, when sitting with crossing legs, someone might prefer to make contact with the chair back surface, while someone else may prefer no back contact. This separation could make the posture expression more accurate and flexible.
- Through the preliminary experiment, if consider the whole body to generate the posture, then one sample of input data includes eight channels, and it relatively effected the whole classification results. For example, if consider the body as a whole, and collect the data for two sitting postures: crossing leg and sitting forward, in the collection stage, some subjects prefer to contact the back surface of the chair in crossing leg posture while the sitting forward without chair back surface contact. One potential situation would happen in the classification stage is that when a subject is sitting on the chair with leg crossed but without back

surface contact, the trained classifier would miss-classified this postures as sitting forward because no back surface contact is a significant feature for sitting forward posture. While after the separation, both body posture classifier and leg posture classifier only need to deal with 4-dimensional data, and this could improve the classification performance.

Another expectation for this experiment is to compare the classification performance across different subjects, not only the performance when dealing with "familiar" subjects, but also the performance when dealing with "unfamiliar" subjects.

So the task of this experiment includes:

- Collect the training data, which includes the measurement from sensors and its corresponding posture label.
- Repeat the data collection process across multiple participants. This task is to make sure the training dataset covers sitting posture data from different subjects.
- Train and evaluate the classification performance of Support Vector Machine based classifier (see section 3.2.4) by using the data from different subjects.
- Train and Evaluate the classification performance when the classifier

deals with data from “unfamiliar” subject.

The experiment detail and result is described in section 4.5.

3.3.5 The Experimental design for Activity Modelling

This experiment’s purpose is to answer research question 5 (objective 3): How to build up the correspondence between the users’ sitting posture and the users’ activity? An important objective of this thesis is to build up the correspondence between sitting posture and sitting activities. Similar to the posture training data in the previous experiment, a training dataset is also required for the activity modelling with the pressure data from the IntelliChair system.

The elements in activity modelling training data are:

- Current timestamp.
- Pressure data from all eight sensors of IntelliChair system.
- Current activity.

And the reason is descried in section 3.2.5.

For the purpose of minimise intrusion to the subjects’ normal activity routine, there is a running GUI program to interact with subjects during the experiment process. The GUI program has pre-defined some activities which are: relax, vacant chair, reading, working with PC, watching video, play games eating, writing.

One thing to emphasize in this experiment is that the experiment tries to investigate more natural activity routine of the subjects, the subjects in the experiment is not required to perform every activities but their natural activities, if they found their current activity is not in the pre-defined activity list, they can use the GUI program to input their own activities into the activity list. Not all the activities are modelled, only the activities with most coverage for all subjects are modelled. The detail and result of this experiment is described in Chapter 5.

3.4 Summary

In chapter 3, author of this thesis concentrates the previous research experience that is discussed in chapter 2, and proposed IntelliChair system as a new approach for sitting posture classification and sitting activity recognition by using non-intrusive, lost cost sensors.

Initially, the local-remote separated system architecture is proposed by author as a baseline according to the previous research. Then, both hardware and software requirement of the IntelliChair are listed by. Author then described the middleware selection as hardware building process of the IntelliChair system. There are two versions of middleware in this research; one is the Arduino version which is used for pressure sensor integration with Kinect sensor. The other one is the Raspberry Pi version which is the main middleware architecture that throughout the rest of the thesis.

Based on the IntelliChair architecture, the discussion is expanded according to five objectives that are listed in section 1.1. Five research questions are listed correlate with the five objectives, which are:

1. (Objective 1a): What type of pressure sensor should the system use?
2. (Objective 1b): Are there any alternative sensors to achieve non-

intrusive detection other than pressure sensing?

3. (Objective 1c): What is the signal characteristic of the pressure sensing information?
4. (Objective 2): How to classify the data in order to detect the sitting posture?
5. (Objective 3): How to build up the correspondence between the users' sitting posture and the users' activity?

For the first question, author narrows the sensor selection options into two (Conductive foam and FSR) by comparing the sensor technologies from section 2.2.1 to section 2.2.3. In order to choose the suitable one, an experiment is designed for the selection process based on the sensor requirement that is described in section 3.1.1.

Author explores the possibility to integrate the visual based sensor (Kinect) along with pressure sensor based on the flexible IntelliChair middleware design (section 3.2.2). In the second experiment about sensor integration, author attempts to visualise both pressure sensor and Kinect sensor data and merge them into one system to convey the sitting posture information to IntelliChair user (section 3.3.2).

The third experiment is designed for the determination of the IntelliChair sampling rate that is used for Activity data collection for the fifth experiment (section 3.2.3). And this experiment could also discover the signal characteristic of posture signal. The experiment design uses only pressure sensing data and implements the Fourier Transform that is usually used in Sampling Theorem to analysis the posture signal in frequency domain (section 3.3.3).

According to the literature review, the fourth (section 3.3.4) and fifth (section 3.3.5) experiment is designed for the estimation of training dataset for posture classification and activity recognition. Because of the different training set labelling, the collection process, GUI for experiment participants is different as well.

For the sitting posture data, author pre-defined 16 sitting posture according to the previous research, furthermore, proposed a new approach which is separate the spine posture and leg posture, which makes the description of whole body posture more precisely.

Because the activity data collection involves the time factor, so the timestamp is recorded as well. In order to investigate more natural activity routine of the experiment participants, a flexible GUI is designed for them to have more activity options or even create their own activities. The experiment results is shown and discussed in chapter 4 (experiment 1 to 4) and chapter 5 (experiment 5).

Chapter 4 Experiment Results

The content of this chapter includes the detail and results of first four experiments that are discussed in section 3.3.1 to 3.3.4. The first four experiments include:

- Experiments for Sensor Selection.
- Multiple Sensor Integration Experiment.
- Experiment for Posture Signal Characteristic Analysis.
- The Experiment of sitting Posture Classification.

Each of the experiment aims to answer one of the research questions discussed in sections 3.2.1 to 3.2.4.

4.1 Experiments for Sensor Selection

Since the objective of the experiment is to evaluate the usability of a pressure sensing material (conductive foam) and the performance of force sensing resistor sensor (FSR), this experiment consists of two parts. The first part is the assessment of the conductive foam and the second part is the assessment of the FSR sensor.

The principle of the performance assessment is to inspect the correlation between the loaded weight and the resistance measurement because the sensing material and the sensors are using resistance as the force indicator. So the method is to apply the weight on to the conductive foam and the FSR and record the correlated resistance. In order to examine their measurement stability and consider the measurement drifting problem mentioned in section 3.3.1, the weight applied on the sensing material and FSR sensor includes two stages:

- Weight increase stage. The weight starts at 0KG (None) increasing to 8KG, in steps of 1KG.
- Weight decrease stage. The weight starts at 8KG decreasing to 0KG (None), in steps of 1KG.

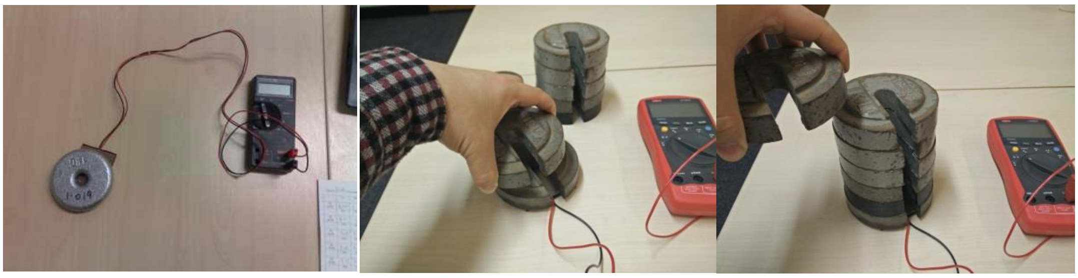
This weight range (0 to 8KG) is selected based on the sensitivity range of both conductive foam and FSR sensor. According to Brady's research (Brady, Diamond and Lau 2005), the maximum of weight that has been loaded onto the foam material is 27.5N (Newton) which is equal to 2.8KG. On the other hand, the FSR sensor's sensitivity range is from 0 to 10KG (Interlink Electronics 2006, pp 5), but according to the Resistance vs. Force plot, after 8KG, the resistance yields very little decrease (the decreased resistance value is only few hundreds compare with K ohms changing in the lighter weight). In order to challenge the potential of the conductive foam as a sensing material and compare their performance in the same condition, the weight range (0 to 8KG) is decided.

The weight changing cycle is applied to both conductive foam and the FSR sensor. The experiment result for conductive foam will be described in section 4.1.1 and the experiment result for FSR sensor will be described in section 4.1.2. There will be an additional experiment for FSR is also included in section4.1.2; its objective is to further explore the measurement feature when heavier weight is applied on the FSR sensor.

4.1.1 The Experiment on Conductive Foam

The experiment equipment for conductive foam resistance measurement includes:

- Different weights: the weights with different amount that can be combined into different loads ranging from 0KG to maximum of 8KG. The weights are manually applied to the foam surface.
- Two copper electrodes: Two copper sheets; each of the sheets is connect with a wire and the wires are connected to multimeters.
- Digital multimeter: The multimeter is utilized to measure the resistance and the measurement is logged manually.



(a) (b) (c)

Figure 4-1 The experiment setup for conductive foam. (a) shows the basic experiment setup, the loaded weight is 1KG. (b) shows the weight increasing stage, in this stage, 1 KG weight is added on top of another weight one by one. (c) shows the weight decreasing stage, in this stage, 1 KG weight is removed from the top one by one. The weight does not have any support the table except the electrode covered conductive foam.

Figure 4-1 shows the experiment equipment setup for conductive foam, and how the weight increasing and decreasing stage is procced. The experiment procedure is to record the resistance measurement 15 seconds later after each new weight is loaded. After the measurement is recorded, the weight will be changed to next weight in steps of 1KG. The reason for the 15 seconds delay is because the resistance measurement is stabilized after a short period of time (around 15 to 20 seconds) This feature and the time delay for measurement is described in Brady's research (Brady, Diamond and Lau 2005), and the oscilloscope plot about its response to the pressure could be found in this Brady's paper. The reason for this feature is because the foam itself is being a light-weight sponge like material, which is sensitive to external pressure. The foam changes into a denser material and the compression effect is less significant, which causes the foam is able to response the pressure quickly but stability of measurement for certain weight takes seconds sometime.

The weight changing follows the weight changing cycle and includes the weight increase stage and weight decrease stage. The cycle starts at 0KG and ends at 0KG; the maximum of the weight increase stage is 8 KG, and this is also the starting point of weight decrease stage. The cycle carries on without stop, and one cycle is a test dataset. The cycle is repeated five times and the record is shown in Table 4-1.

Ω (kΩ)	Loaded Weight	None	1KG	2KG	3KG	4 KG	5 KG	6 KG	7 KG	8 KG
Test 1 (Weight Increase) →		252.6	22.69	2.624	0.963	0.635	0.475	0.401	0.332	0.293 ↓
Test 1 (Weight Decrease) ←		494	2.219	0.678	0.456	0.366	0.324	0.298	0.283	0.293 ←
Test 2 (Weight Increase) →		224.3	23.61	1.892	0.887	0.632	0.498	0.393	0.326	0.288 ↓
Test 2 (Weight Decrease) ←		378	2.385	0.708	0.468	0.372	0.3224	0.2918	0.2796	0.288 ←
Test 3 (Weight Increase) →		123.2	22.34	1.561	0.722	0.487	0.387	0.328	0.305	0.289 ↓
Test 3 (Weight Decrease) ←		507	1.703	0.525	0.385	0.327	0.303	0.2813	0.2762	0.289 ←
Test 4 (Weight Increase) →		126.8	30.3	1.534	0.803	0.600	0.513	0.421	0.378	0.342 ↓
Test 4 (Weight Decrease) ←		279.4	1.715	0.62	0.469	0.406	0.370	0.351	0.335	0.342 ←
Test 5 (Weight Increase) →		217.4	21.47	1.540	0.658	0.459	0.364	0.287	0.2387	0.2087 ↓
Test 5 (Weight Decrease) ←		407	1.504	0.531	0.348	0.2723	0.2287	0.2107	0.2011	0.2087 ←

Table 4-1 The measurement record for the conductive experiment. The unit is Kilo Ohm.

The plot that indicate the signal response performance of the conductive foam as a pressure sensing material is shown in Figure 4-2, and each of the plot represent one test dataset. According to table 4-1, the measurement interval from 0KG to 2KG is too large (200 kΩ is significant larger than 2 kΩ) for the rest of the data, so 2KG is selected as the starting point for data plotting. The result confirms that the resistance of the conductive foam is able to change according to the loaded weights within the range of the weight changing cycle. But result also shows that the conductive foam could not maintain the measurement between cycles, which means when the same weight is applied on the conductive foam, the measurement could not maintain in a certain range because the measurement from increasing and decreasing weight diverge, especially when the weight is lighter than 3KG. The data shown in this figure indicate that the conductive foam material's repeatability for resistance on the same weight is poor. The main reason for this is probably because the foam has been subject to compression intensely which cause the loss of internal energy.

This experiment indicates the potential of conductive foam as pressure sensor but it has limitations which are measurement hysteresis after foam compression and low repeatability of measurement.

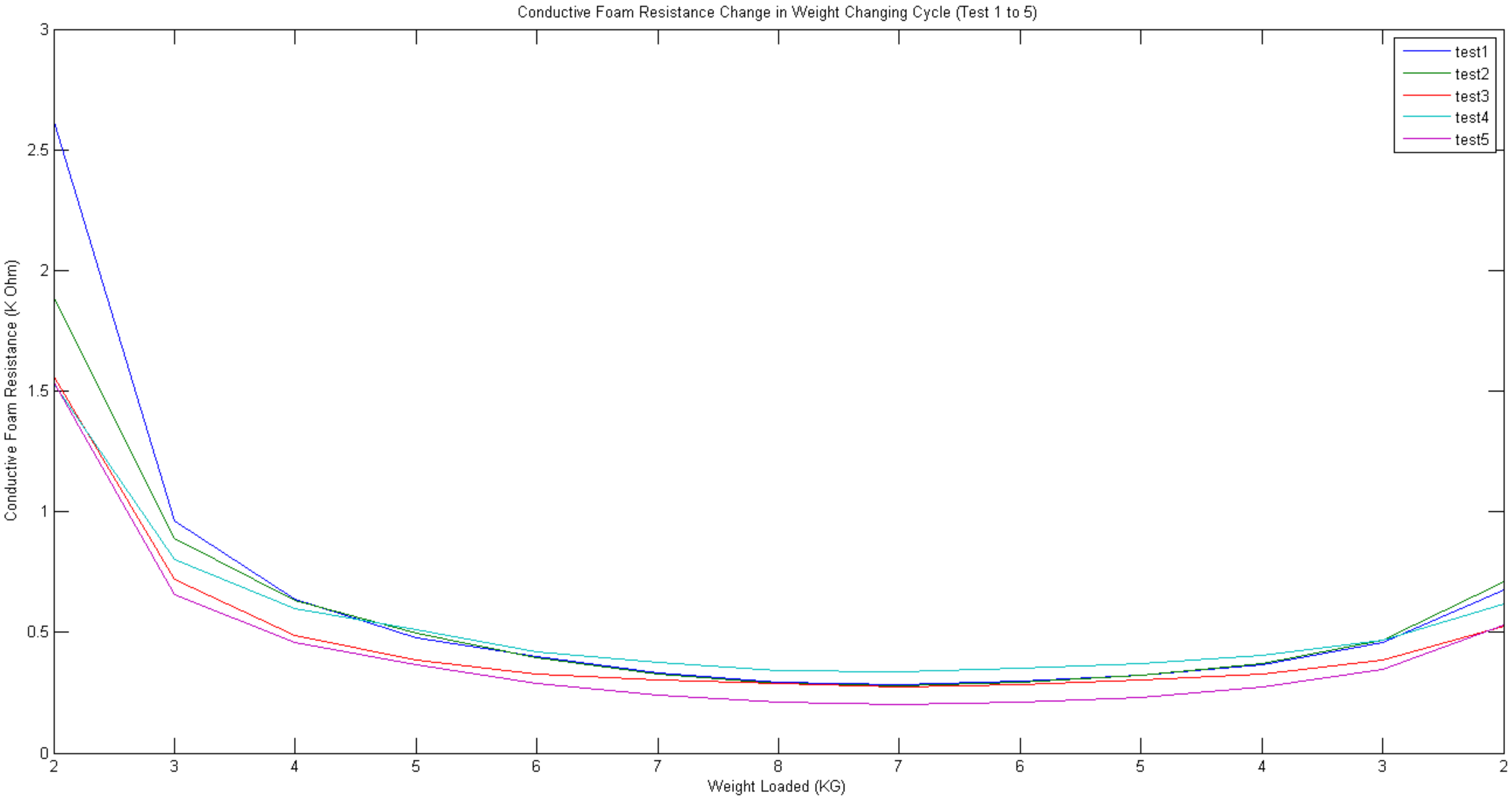


Figure 4-2 The plot of the conductive foam resistance change in weight increase and decrease cycle (Test 1 to 5).

4.1.2 The Experiment on FSR Sensor Resistance

The experiment equipment for FSR sensor resistance measurement includes:

- Different weights: weights with different amounts that can be combined into different loads range from 0KG to maximum of 8KG (for the additional experiment for FSR sensor). The weights are manually applied to the sensor surface.
- Digital multimeter: The multimeter is utilized to measure the resistance and the multimeter is connected to a PC and record the measurement automatically.

The copper electrodes are not required in this experiment because the FSR already

The basic experiment procedure is the same as the experiment on conductive foam; a weight changing cycle is applied on the FSR sensor, except that the measurement is recorded automatically. The reason for the change of automatically record is because according to the experiment on the conductive foam the manually recording of data would cause measurement deviation. That is why the multimeter with measurement recording function is utilized for the experiment on FSR. The weight changing cycle steps by 1KG and includes weight increase and decrease stage with range from 0KG to 8KG. The cycle will repeat three times, and each cycle is one test dataset.

The reason for the three times instead of five times for FSR is according to the initial experiment on FSR, the performance of FSR that maintains the resistance is relatively better than conductive foam.

The reason for the evaluation difference for conductive foam and FSR sensor is because the FSR is a commercialized product which has standardized performance according to its datasheet (Interlink Electronics 2006). So in the experiment for FSR sensor, the times of cycle is changed to 3 times, and the time interval of weight changing is changed to 10 second because the measurement of FSR stabilized more quicker compare with conductive foam.

The weight will change every 10 seconds based on the preliminary experiment experience that the resistance measurement is stabilized around 10 seconds. The sampling rate of the multimeter record function is 1Hz, which means one sample per second. The whole weight increase and decrease sequence generated 170 samples and every 10 seconds correspond to a specific weight, this is listed in table 4-2.

Sample NO.	0-9	10-19	20-29	30-39	40-49
Weight(Increase)	None	1 KG	2 KG	3 KG	4 KG
Sample NO.	50-59	60-69	70-79	**80-89**	
Weight(Increase)	5 KG	6 KG	7 KG	**8KG(Max)**	
Sample NO.	**80-89**	90-99	100-109	110-119	120-129
Weight(Decrease)	**8KG(Max)**	7 KG	6 KG	5 KG	4 KG
Sample NO.	130-139	140-149	150-159	160-170	
Weight(Decrease)	3 KG	2 KG	1 KG	None	

Table 4-2 The sample numbers and their corresponding weight.

Figure 4-3 shows the measurement changes along with the sample numbers. Since the resistance measurement from sample number 0 to 10 (the weight is 0KG) is significant higher than the rest of the measurement, sample number 11 is selected as the starting point for data plotting.

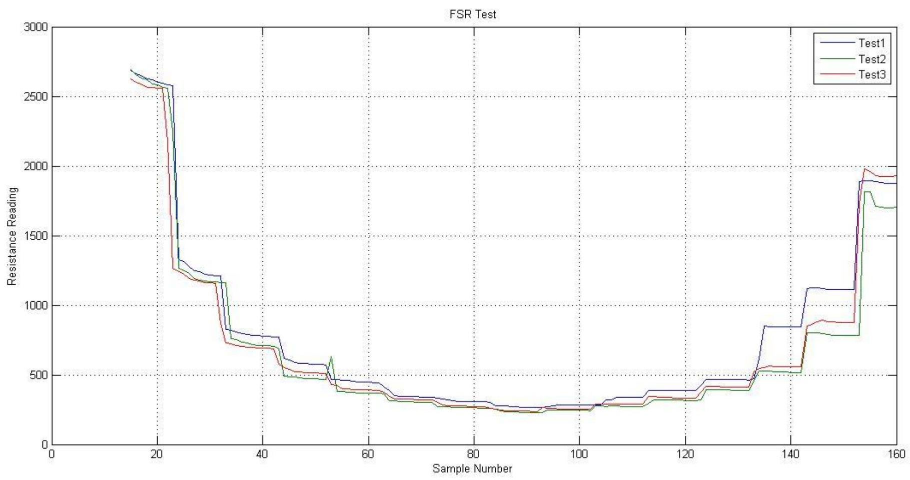


Figure 4-3 The resistance and sample number plot for the three weight increase and decrease sequences. The unit for resistance is ohm

The measurement repeatability of the resistance is relatively stable according to the plot, but it also shows that when the weight is higher than 4KG, the curve become smoother and less response to the increasing weight. Because the better measurement repeatability performance compare with conductive foam in short time period (period lasts tens of seconds), an additional experiment is launched in order to further discover the feature of the FSR based on the above results.

The purpose of the additional experiment is to explore the measurement repeatability performance in the longer term period (period lasts minutes) which is similar to real situation. In order to simulate the real deployment situation, the experiment strategy also changed.

The additional experiment for the FSR firstly narrows the weight increase and decrease range at a high weight (4 KG as the starting point, while 8.5 as the maximum weight), the weight change for each step is 0.5 KG instead of 1KG. The purpose of changing the weight range is because:

- This experiment is only for FSR, so the maximum weight slightly exceeds 8KG (8.5 KG, but still within 10 KG) in order to investigate the FSR's performance in a larger range.
- Decrease of each step is to investigate the FSR's performance in a smaller weight scale.
- Minimum weight starts at 4 KG because the FSR's resistance measurement start to converge into small scale according to Figure 4-3 (units changed from thousands of ohm to hundreds). The reason for not record the measurement before 4KG in this additional experiment is also because the big leaps of scale, for in the real sitting posture classification situation, such big differences of measurement change could be easily recognised by classifier (see section 4.4.4).

Another change is that each weight will stay on the sensor for at least 8 minutes instead of 10 seconds. According to Hermann's research on sitting discomfort (Hermann 2005), people usually changes sitting posture around 10 to 15 minutes. Furthermore, the long term situation is not only helps for sitting posture classification but also the sitting activity recognition stage. In order to simulate the long term weight loading situation, a near 10 minutes (8 minutes) strategy is selected for this additional experiment.

This additional experiment is to see whether FSR sensor is still responsive under those conditions:

- The loaded weight is high.
- The loaded weight stays contact for a long period of time.
- Change of the weight is less significant.

This experiment setup tries to build up a test environment which is similar with real situation in order to better evaluate the FSR's performance.

Because the measurement lasts 8 minutes, the sample numbers are quite large. In order to plot the data, the loaded weights along the x axis correspond with the mean value of the resistance measurement instead of an array of measurements. Table 4-3 shows the correspondence and the plot is in figure 4-4. The weight loading and un-loading procedure is the same with conductive foam experiment which is shown in Figure 4-1; the only difference is the different weight for each step and different staying time of a weight for each step.

Weight (Increase)	4 KG	4.5KG	5 KG	5.5 KG	6 KG	6.5 KG	7 KG	7.5 KG	8 KG	8.5 KG
Ω	→ 1191.920	899.540	648.075	537.155	449.932	387.200	326.146	293.352	265.797	↓ 247.321
Weight (Decrease)	4 KG	4.5KG	5 KG	5.5 KG	6 KG	6.5 KG	7 KG	7.5 KG	8 KG	8.5 KG
Ω	701.229	583.145	468.121	405.579	350.204	316.718	288.0031	269.343	253.855	← 247.321

Table 4-3 The correspondence of the loaded weight and the measurement average value over 8 minutes.

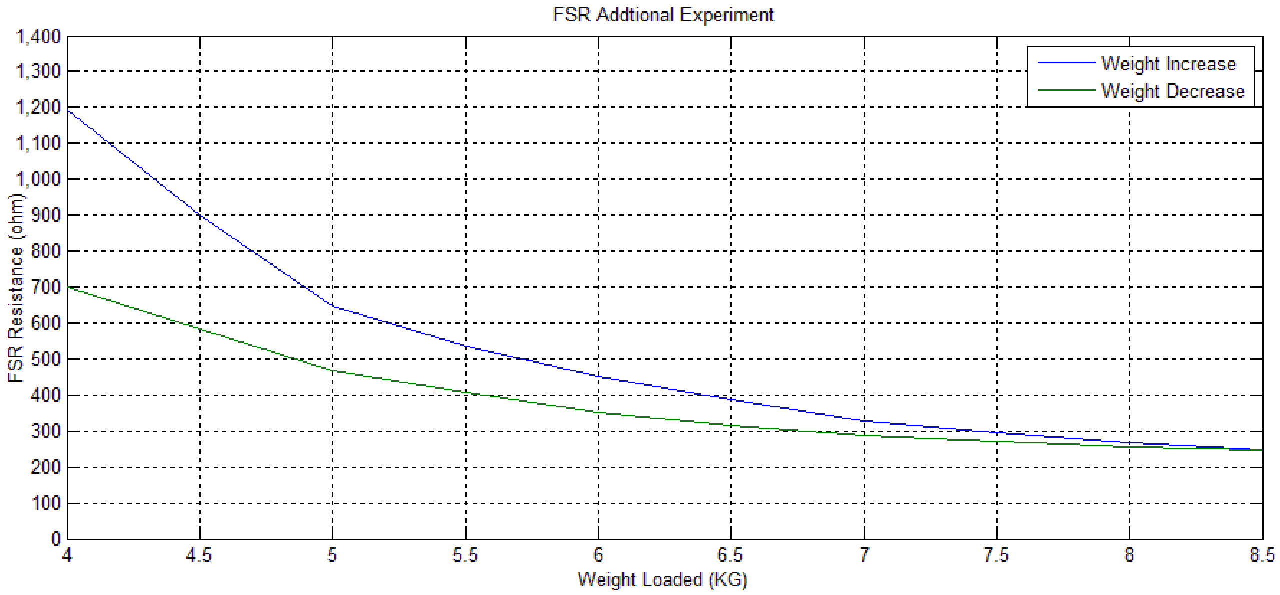


Figure 4-4 The plot of average resistance measurement with loaded weight.

The curve in figure 4-4 indicates that the FSR sensor still able to respond to different weight when high weight is loaded, but this plot is based on the resistance measurement average of 8 minutes windows. Hereby, two questions arise that are:

1. The big difference at 4 KG, does it means FSR is not reliable?
2. If the sensor is still reliable, can the stability of the measurement be maintained within the whole 8 minutes?

In order to answer question 1, the sensing mechanism is the key. The sensing mechanism for conductive foam and FSR are the same, which means both of them have the measurement hysteresis problems. But the difference between them is that the problems even happens in short term (tens of seconds long) situation for conductive foam, while the problems occurs in long term (8 minutes) situation for FSR sensor. The performance of FSR sensor is better than conductive foam in short term by comparing figure 4-3 (FSR) and figure 4-2 (conductive foam). Even in the long term situation, the difference of the measurement is not a major issue because the posture change after long term will come with big pressure release (imaging change sitting posture from sitting straight to cross legs, etc.) which causes big measurement differences, the posture classifier is able to recognise the difference. So the answer for question 1 is positive.

To answer question 2, the data is re-organised, and take time axis into account. Instead of using the average value of the whole 8 minutes measurement, the window size to calculate the average value is 1 minute.

Because time is another factor of the additional experiment, so the time axis is imported in plotting to monitor the measurement changing within 8 minutes. Figure 4-5 and figure 4-6 are 3D plots which indicate the measurement change over time and weights. In figure 4-5 and figure 4-6, the measurement value is visualized by colour, and it is clear that within the 8 minutes time with the same weight applies on the FSR sensor, the colours of each minute is almost the same or with minor changes. The average standard deviation of the measurement is around 2.68%.

This result indicates the measurement of the FSR sensor is relatively stable under the condition of long time and high weight. And the measurement is still responsive to the pressure change step by 0.5KG.

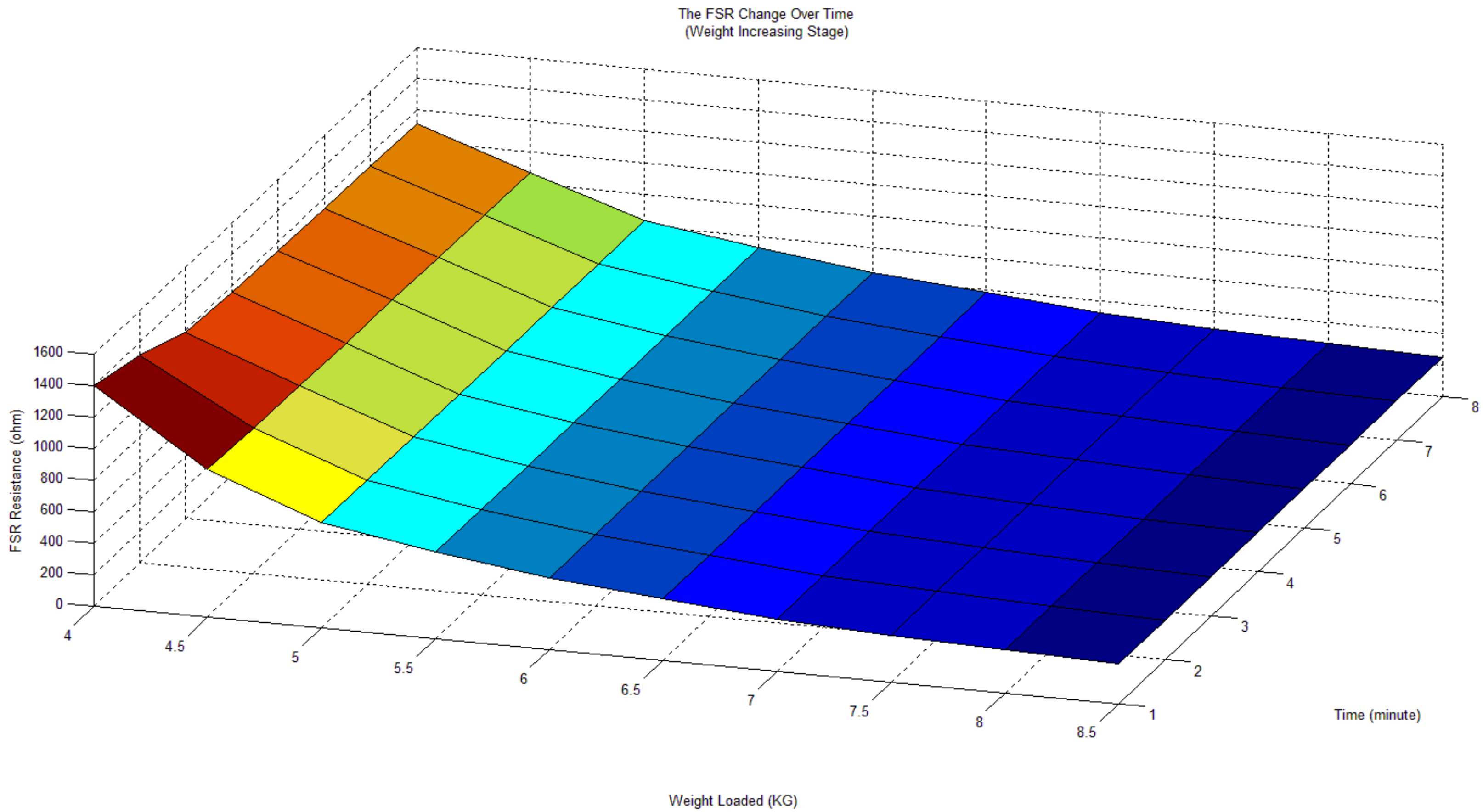


Figure 4-5 Three dimension plot representing the resistance change in weight increasing stage.
This plot includes the resistance measurement mean (Y axis), corresponding loaded weight(X axis) along with the time (Z axis). The colour indicates the value of the resistance.

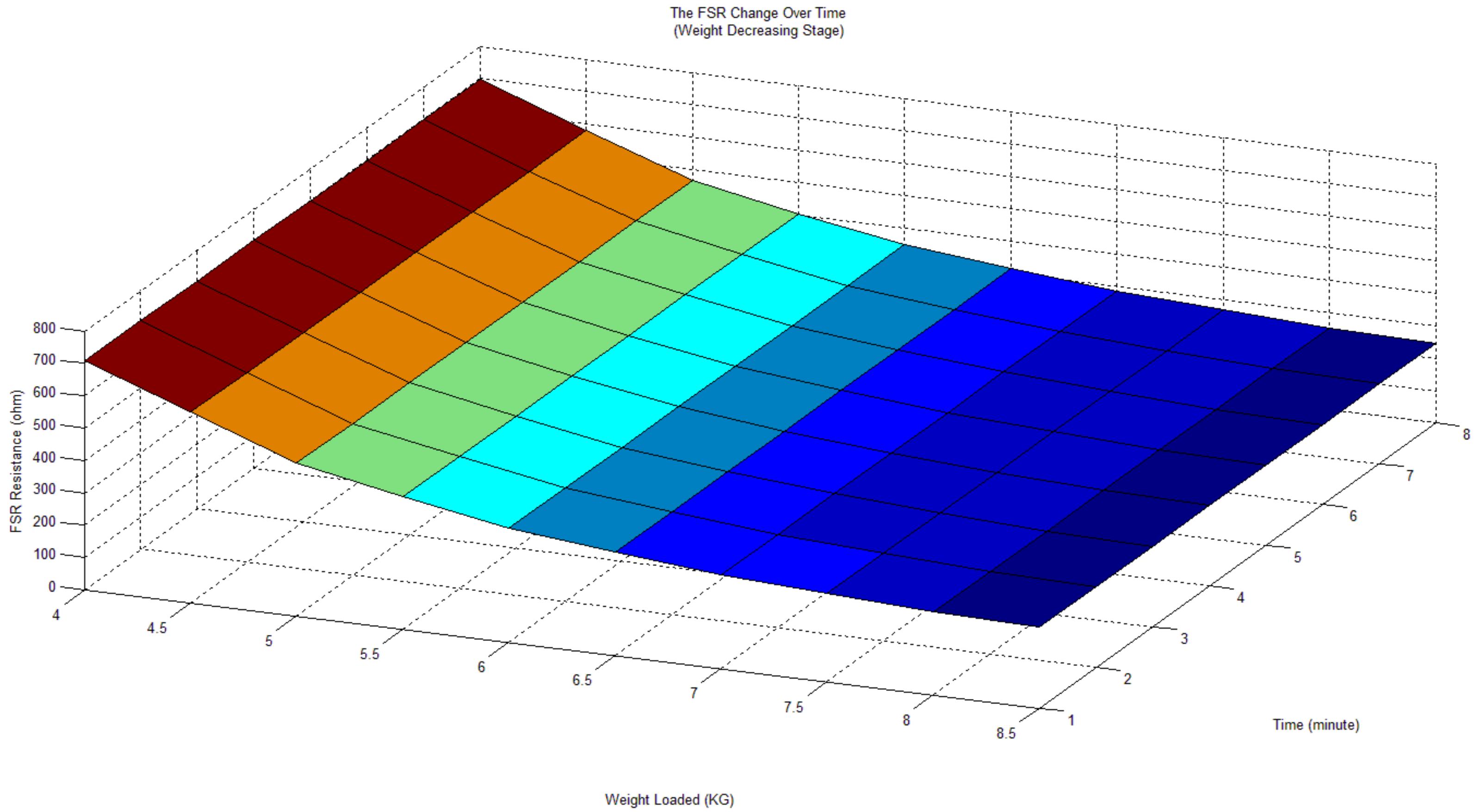


Figure 4-6 Three dimension plot representing the resistance change in weight decreasing stage.
This plot includes the resistance measurement mean (Y axis), corresponding loaded weight(X axis) along with the time (Z axis). The colour indicates the value of the resistance.

4.1.3 Summary of Sensor Selection Experiment

According to the discussion of the experiment result in section 4.1.1 and section 4.1.2, it can be figured that there are limitations for both conductive foam and FSR sensor. They both have quick response to the pressure change within seconds (as shown in Figure 4-3), but they also both have the measurement drifting problems or measurement repeatability.

As discussed in the end of section 4.1.2, the conductive foam could not maintain a relatively stable resistance measurement in the repeating weight increase and decrease process compare with FSR in short term condition. FSR does also have repeatability problem, but it occurs in the long term weight applied situation. Furthermore, according to the FSR integration guide (Interlink Electronics 2006), the effect of the measurement drifting problem could be decreased by pre-load weight on the FSR and then begin the usage. Meanwhile, FSR sensor is able to detect minor pressure changes even in the long term situation. Refer to Figure 4-5 and Figure 4-6, the measurement for same weight stays in the same colour in 8 minutes of measurement recording and there is clear difference between different weights. This means that FSR is capable of capturing minor pressure difference.

Through the results and discussion, the FSR sensor is chosen to be the pressure sensing unit of the IntelliChair system, the reasons are:

1. The FSR sensor provides lower measurement deviation according to the result.
2. The FSR sensor is able to maintain its measurement stability under the long time and high pressure condition, meanwhile the FSR sensor is still responsive to the minor pressure changes. This means in the real situation, the FSR sensor should be capable of capturing even minor pressure differences between different sitting postures.

The objective of the sensor selection experiment is achieved and the research question 1 (What type of pressure sensor should the system use?) for objective (1a) is answered.

4.2 Multiple Sensor Integration Experiment

The aim of this experiment is to validate the usage and the reliability of Kinect sensor data with pressure sensor data by visualizing them together in real time. Refer to the experimental design in section 3.3.2, the data visualization component consist of two sub-components:

- Visualization of the Pressure Data from FSR sensors. This component visualizes the pressure sensor data along with the placement of the sensors.
- Visualization of the Skeleton Data from Kinect sensor. This component delivers a 3D based humanoid object to represent the human sitting posture.

Furthermore, the two sub-components are developed to be capable of working either individually or together. This modular development helps the assessment of the usability for sitting posture detection when they are working standalone. If two components are working together, the overall system aims to convey visual based information to the system users which helps them better understand the sitting posture when they are using IntelliChair.

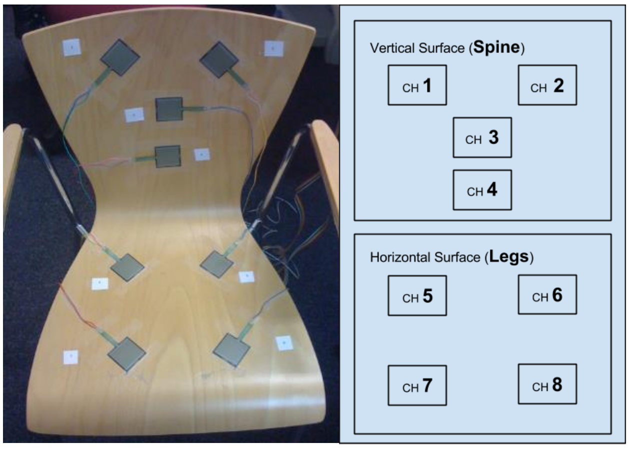
When the two sub-components are integrated, the overall system acquires the skeleton data first and then the pressure data for synchronization purpose.

The reason for that is because the data size of the skeleton (an array of 5*20) is bigger than pressure data (an array of 1*8), and dealing with pressure data later will make the whole system display the information near real-time.

4.2.1 Visualization of the Pressure Data

The pressure data visualization sub-component visualizes both the pressure data and the sensor placement, so this component is dependent on the IntelliChair pressure sensing hardware architecture which is shown in Figure 4-7. In this figure, eight FSR sensors are mounted on the chair. Four units are on the horizontal surface and the rest are on the vertical surface. On top of the sensors, there is a layer of polymer mat, between sensor and human body.

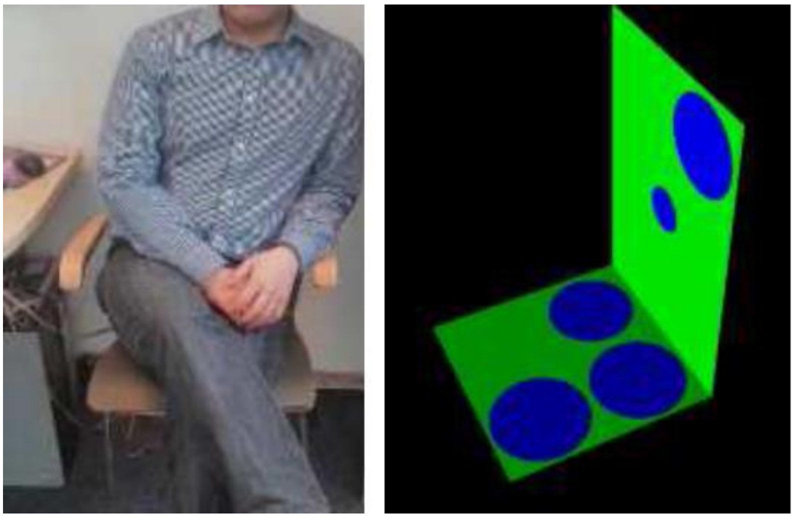
The reason for this sensor installation strategy is because fixing the FSR sensor on a hard surface could make the pressure measurement of the FSR more stable. Another reason is the protection of the sensor. The FSR sensor has certain flexibility, but if they are deployed on a cushioned chair, the deformation of the chair surface when pressure is applied could potentially damage the sensor (deformation of the sensor itself, causes unreliable measurement), especially the pressure is going to be applied for hours long. With a polymer mat on top of the sensor, the system could have stable pressure measurement, meanwhile protects the sensor from deformation.



(a) (b)

Figure 4-7 The sensor placement and correlates channel numbers.
(a) Chair with FSR mounted sensor positions.
(b) FSRs are represented by channel numbers that is correlated with (a).

Another important feature for pressure data visualization is to convey the pressure from each FSR sensor (channel). In this component, each FSR is incorporated in a potential divider circuit and its resulting voltage is digitized as a 10-bit numeric (0 to 1023). The radius of a circle that is shown in Figure 4-8 is the logarithm of the 10-bit numeric to the base 2. The larger the circle area is, the heavier the pressure is which is shown in Figure 4-8.



(a) (b)

Figure 4-8 Demonstration of pressure data visualization component.
(a) is the sitting posture while (b) is the real time display of the force Information, the circle area indicates the pressure.

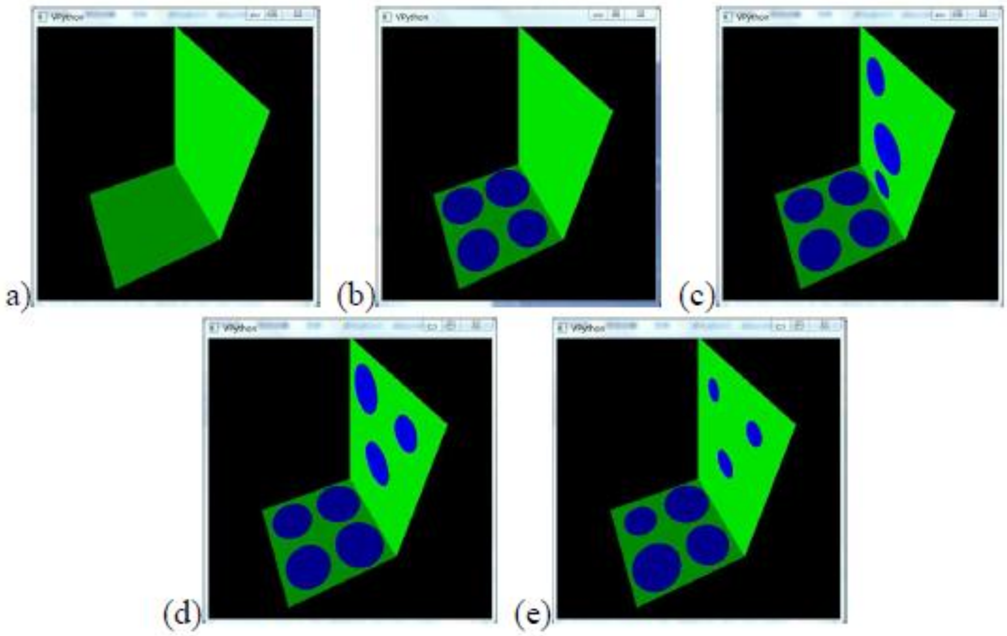


Figure 4-9 3D visualization of pressure data of different sitting postures.
(a)Nobody (b) Sitting upright (c) Body Lean Right (d) Relaxing Back (e) Body Crouch.

Figure 4-9 demonstrates that this component is able to display the pressure distribution information generated by different sitting postures, and it could convey to the user the current sitting posture that corresponds with current the pressure distribution.

The visualization is 3D based because the skeleton data from the Kinect sensor is an array of 3D data which pin-point the specific joint within a 3D space, so a 3D visual space is easier for the integration for both data.

The pressure data visualization component achieves the requirement of real time data visualization it helps the IntelliChair user to directly understand the pressure distribution of a posture.

4.2.2 Visualization of the Skeleton Data

The skeleton data visualization sub-component visualizes the 3D based position data of 20 key joints (as shown in Figure 4-10) of the human skeleton. By assembling those joint, a 3D based humanoid object is created to convey human posture information.

The Kinect sensor, as introduced in section 2.2.4, captures the depth information through its camera and the Kinect sensor SDK is able to estimate the 3D based position data of 20 joints in a virtual, Kinect-cantered, 3D space.

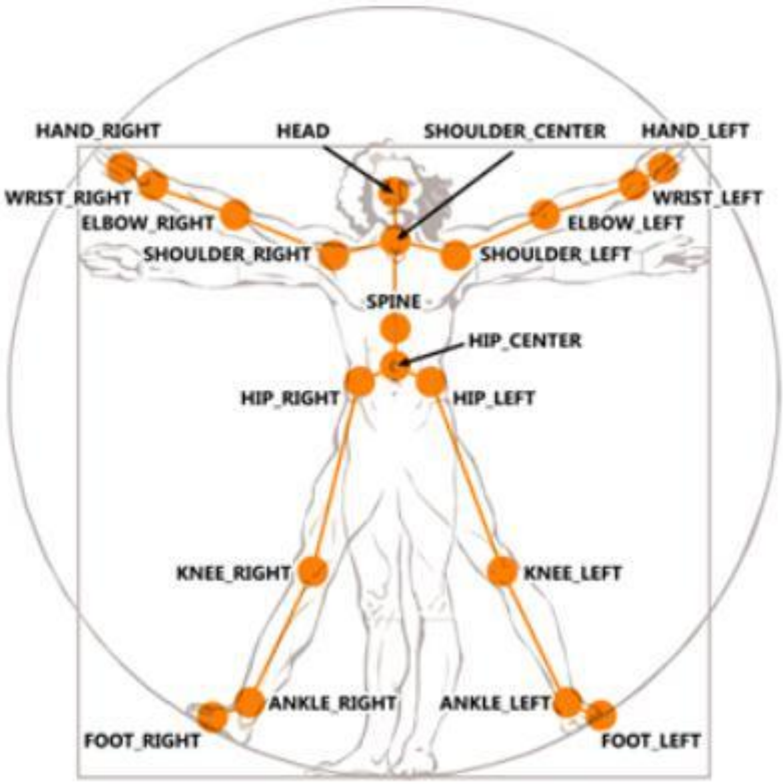
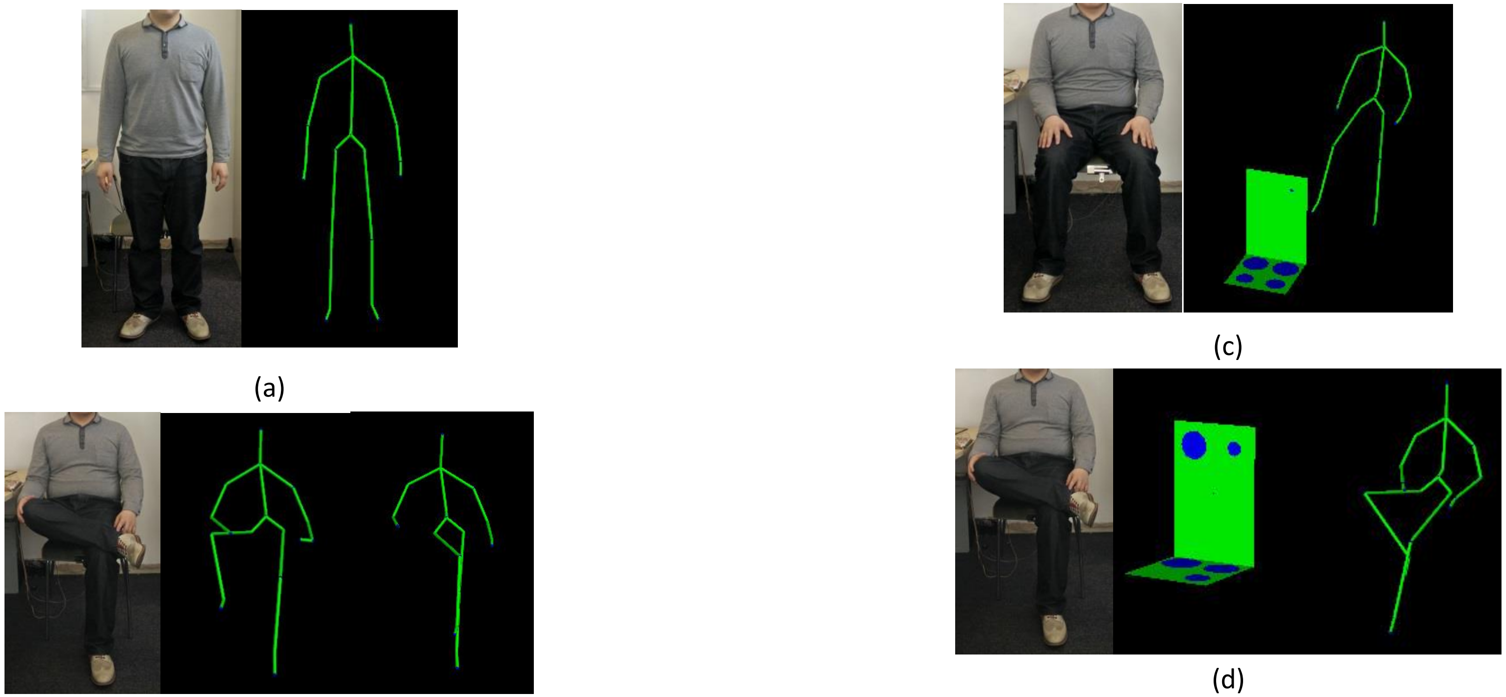


Figure 4-10 The joints that are estimated by the Kinect SDK (Holmquest June 2012).

This component conveys the human posture more easily, and more understandable for user because it creates a humanoid object which is directly related to the user’s posture as shown in figure 4-11. With 30 FPS sampling frequency, the humanoid object also has a quick response to the user’s posture change.



(a) (b) (c) (d)

Figure 4-11 3D interface of skeleton data from Kinect sensor and pressure data from IntelliChair.

(a) Standing in front of Kinect sensor, no joint overlap, and skeleton stable.

(b) Sitting in front of Kinect sensor with leg crossed, the posture is stable, but the two skeleton images from Kinect do not reflect the posture correctly. The right leg in the images is not in a correct position compare with subject's posture, and it means the joint drifting problem occurs.

(c) Synchronised pressure and skeleton data for sitting straight posture.

(d) Synchronised pressure and skeleton data for crossing leg. Right leg is not in the correct position, joint drifting still exists.

There is a problem for this component which is the joint drifting problem which is also indicted in Figure 4-11 (a) and (b). The joint drifting problem is that Kinect sensor established inaccurate or wrong joints 3D coordinate when sensed human subject is in front of Kinect sensor and subject is stationary, and those fault coordinates cause the visualisation component convey the wrong human posture. As described in section 2.2.4, this problem is caused by the Kinect sensor because of its depth information based nature. Kinect sensor requires correct positioning in front the user to ensure its performance because Kinect sensor predicts each joint's position data in a Kinect-centred 3D space.

Although all the joints are estimated, there are differences among them. In the Kinect sensor SDK, each joint has a property called JointTrackingState, which has three possible values:

- Tracked: It means the joint is detected directly from depth image.
- Inferred: The joint point could not be seen from depth image, but it could be predicted by considering other related joints.
- NotTracked: The joint point could not be determined; the reason causing this may be that the joint is out of the detection range.

In the real situation, the joints with *Tracked* label are reliable because it is the part that has been directly abstracted from depth image, while the Inferred ones are frequently fails due to occlusions (e.g. self-occlusion by other body parts, e.g. in sitting postures, the hip joints are usually blocked by knee joints). This failure of joint position prediction causes the joint drifting problem, which is shown in Figure 4-11 (the left leg is shorter than in the right leg, which means the incorrect estimation of leg knee joint from Kinect).

In the experiment, the joint drifting problem is assessed in the sitting posture monitoring, but the assessment is not based on quantified information but from anecdotal feeling from the thesis author. This anecdotal feeling has also been described in Kinect support documents from Microsoft. In the Human Interface Guidelines (Microsoft 2015), it address the joints drifting problem when Kinect is monitoring human sitting situation, but using the following description: “Kinect for Windows can also track seated skeletons with only the upper 10 joints in the seated mode”.

Researchers from HCI also addresses the drawback of Kinect and concludes that Kinect skeleton tracking struggles with occluding body parts or objects in the scene” (Obdrzalek et al. 2012). From the user experience in the experiment and research results in the literature (Obdrzalek et al. 2012), the Kinect sensor does not have reliable performance in sitting posture detection.

4.2.3 Summary of Multiple Sensor Integration Experiment

The multiple sensor integration experiment achieves the two objectives in section 3.2.2 which are: Evaluate the usability of the Kinect sensor for sitting posture detection. Then, explore the possibility to integrate the Kinect sensor data with pressure sensor data by visualize them together in real time.

Firstly, the result shows that the IntelliChair system is able to synchronize and visualize the data from FSR sensor and Kinect sensor for information display in a virtual 3D space. Secondly, the modular development enables two components to run individually, so the elements including the accuracy of data display and response to the posture change can be evaluated individually.

According to the experiment results as shown in Figure 4-11 (c) and (d), the pressure sensing component is more reliable because it can visualize the pressure distribution information accurately and quickly response to the user's sitting posture change.

On the other hand, Kinect sensor does not have a reliable performance in sitting posture detection because of its depth information based sensing nature. This nature causes the joint drifting problem which occurs in the lower part of the body (hip, thigh, leg, and knee). This is not acceptable for system that aims to detect sitting posture because in sitting situation, several joints (knees and hips) appear overlapped in the depth information.

In conclusion, for IntelliChair research, the focus will concentrate on pressure sensing based sitting posture detection due to its reliable performance. This is answer to research question 2 (Are there any alternative sensors to achieve non-intrusive detection other than pressure sensing?) for objective (1b).

4.3 Experiment for Posture Signal Characteristic Analysis

The aim of this experiment is to find out the most suitable sampling frequency rate for IntelliChair system to collect the pressure information. In order to determine the suitable sampling frequency, the pressure information which is generated by human sitting posture is considered as a signal and by analysing its frequency characteristic, the suitable sampling frequency rate can be determined.

The basic experiment procedure is:

1. Participant sits on the IntelliChair and makes frequent body and chair surface contact.
2. The pressure data (voltage) from one FSR is recorded along with timestamp. The reason for recording voltage instead of resistance is because in this experiment, the IntelliChair system is connected to support circuit and the resistance of FSR sensor is converted to voltage.
3. The time-series pressure data is converted to frequency-domain data by using Fourier Transform.
4. The frequency-domain data is drawn in plot and find out the least significant peak of the curve determined. The reason to find out the

least peak is described in section 3.3.3. Because the Nyquist rate is twice the least peak frequency.

The theory behind the experiment refers to the signal re-construction theory which is discussed in section 3.3.3. The least significant peak of the curve in the plot means the maximum frequency responses of a signal. If the sampling frequency of a system is higher than twice this maximum frequency (Nyquist Rate), the system could re-construct the signal without loss of any major information. In this research, it means if the sampling frequency of the IntelliChair system is higher than the Nyquist Rate of the sitting posture signal, the IntelliChair system will not miss any critical change of posture information.

This experiment consist of two different data groups; the first data group is collected based on the relatively lower sampling frequency (75.4Hz and 81.4Hz, because the IntelliChair system is under development in this stage) of IntelliChair system. It contains two sets of data, and each set is collected when the subject is sitting on the IntelliChair and shaking his leg quickly which causes frequent contact between his thigh and the flat chair surface. The IntelliChair records the voltage change of one channel which corresponding to the leg that is shaking. The first data group will be used to establish the basic signal characteristic of the human sitting posture.

After the improvement of the IntelliChair software which aims to increase the overall sampling frequency with all eight sensors, the second data group is collected with higher sampling rate (range from 190.6Hz to 198.9Hz). There are five sets of data in this data group; each of them represents a different pattern of micro motion from different body segment. All the dataset that are collected in this experiment is from one individual subject (author of this thesis).

Dataset Number	Body Segment	Micro Motion Level	Average Sampling Frequency	Channel
1	Right Thigh	Very quick shaking	197.5	7
2	Right Thigh	Quick shaking	197.4	7
3	Right Thigh	Normal Contact	191.7	7
4	Right shoulder	Normal Contact	198.9	1
5	Right Hip	Normal Contact	190.6	5

Table 4-4 The different situation that each dataset represents.

Table 4-4 shows the body micro motion pattern observed and the correlated body segment. The level of the micro motion represents how frequent the body contacts the chair surface and their differences are as follow:

- Very quick shaking means that the subject makes the best effort to maximizes the contact frequency with the chair surface which simulates the situation that the person who is sitting on the chair is in a stroke condition.
- Normal Contact means the subject is sitting normally on the chair, and performs normal postures changing like from sitting upright to relaxing back.
- Quick shaking is a level that in between those two level. It means the subject is contacting the chair surface frequently, but not as frequently compared with the very quick shaking level. This is the simulation that the person who is sitting on the chair has the habit that shaking the legs (e.g. by tapping the floor) when feeling nervous, but does not reach stroke condition level.

From experience, leg and thigh can generate the maximum body and chair contact frequency, while in most of the time, the upper of the human body stays in normal contact condition when the subject is sitting. That is the reason why in this experiment, the very quick shaking and quick shaking pattern is only performed on the thigh, because even if the shoulder or hip performed the extreme pattern, the signal frequency will not exceed the frequency that is generated from the subject’s thigh. In order to simplify the experiment process, only right part of the human body signal is recorded.

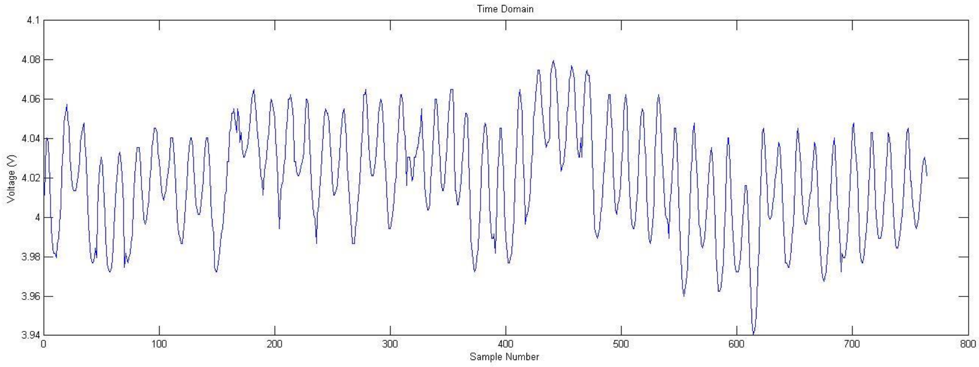
4.3.1 Frequency Domain Analysis on First Data Group

The content in this section describes the experiment on the first data group where the data is collected at a relatively lower sampling frequency (75.4Hz and 81.4Hz). The data collection and analysis procedure is described in section 3.3.3 which includes:

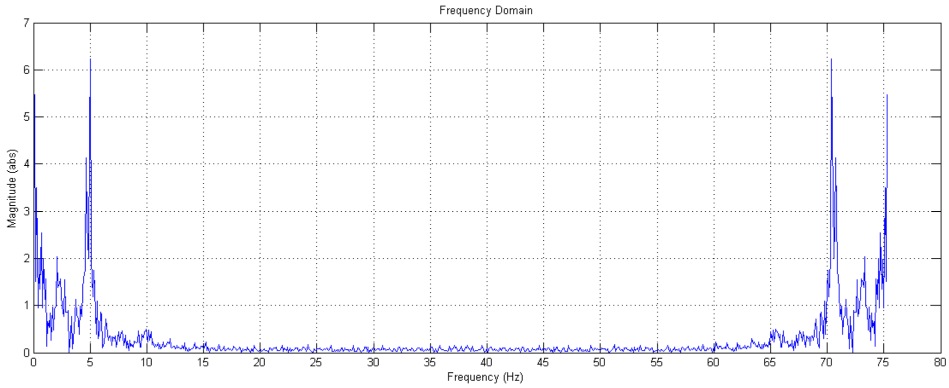
1. Human subject is asked to sit on the IntelliChair and perform quick shaking human-chair contact pattern for 10 seconds.
2. The pressure data, which is the voltage from the sensor will be recorded in time order.
3. After the data collection, the dataset is converted into an array by using Python Numpy package (Numpy Developers 2013), and because the Fourier transform function is also integrated in the package, so by invoking the fft() function (a function that perform the Fourier Transform), the whole array is converted into frequency domain information.
4. The Magnitude and Frequency plot is drawn in Matlab, to find the peaks to determine the maximum frequency.
5. Determine the Nyquist Rate for each human-chair contact pattern by using the maximum frequency from the previous step (step 4).

The data is plotted in both time domain (Figures 4-12 a and 4-13 a) and frequency domain (Figures 4-12 b,c and 4-13 b,c) for better understanding of the signal features in this section. Figure 4-12 b and Figure 4-13 b covers the whole frequency range, while Figure 4-12 c and Figure 4-13 c covers the first half of the frequency range.

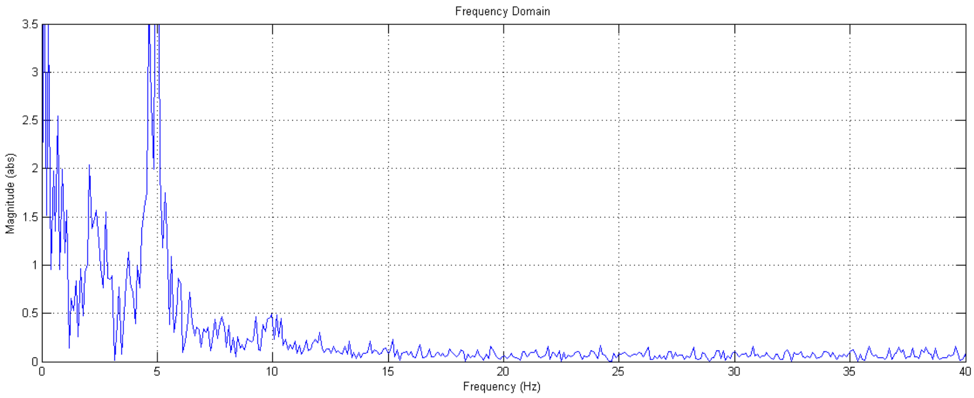
The least significant peak should be determined in Figure 4-12 c and Figure 4-13 c instead of full frequency range. The reason for this is because of the feature of Fourier Transform. After the time domain information is transformed into frequency domain by Fourier Transform, the information in the frequency domain exhibits mirror symmetry with respect to the half of the full frequency (37.7 Hz in Figure 4-12 and 40.7 Hz in Figure 4-13). The mirror symmetry feature of the frequency domain plot is caused by the nature Fourier Transform. Because the Fourier Transform reveals the frequency features of a signal by breaking it up into complex exponentials, and a sine wave is the sum of 2 complex exponentials, which causes the mirror symmetry.



(a)

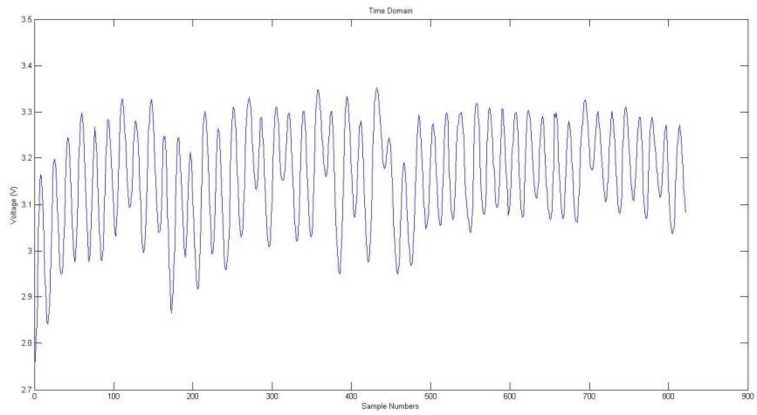


(b)

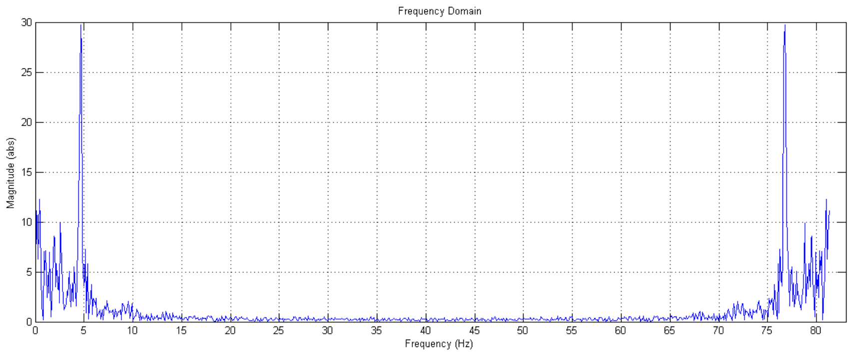


(c)

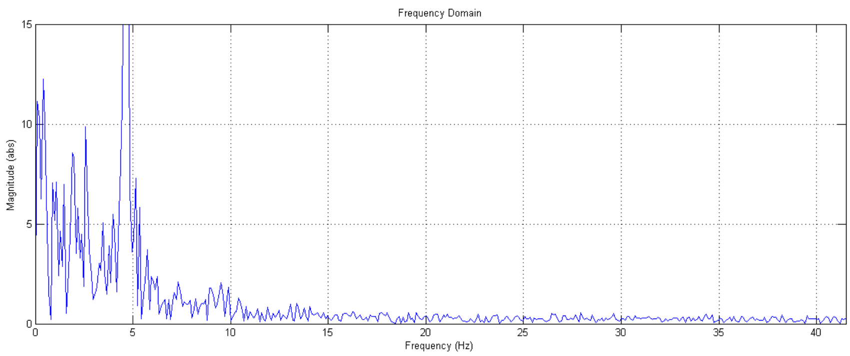
Figure 4-12 The plot for dataset 1 in group one (quick shaking pattern).
(a) represents the data in time-domain, while (b) and (c) is the data in frequency domain, (b) displays the frequency information in the full frequency range (0-75.4 Hz), (c) is a zoom in version of (b) which focuses on the half range of the frequency (0-40 Hz).



(a)



(b)



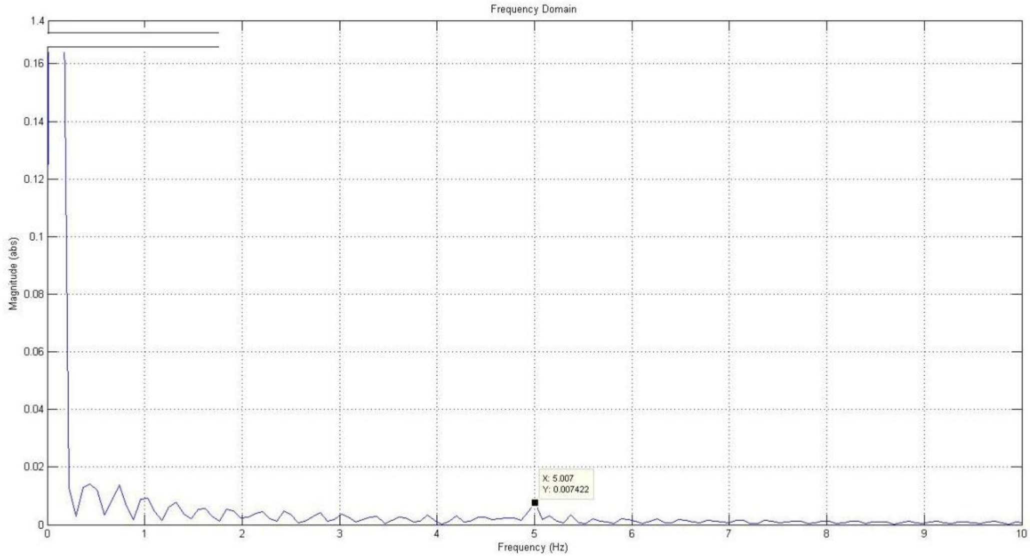
(c)

Figure 4-13 The plot for dataset 2 in group one (quick shaking pattern).
(a) represents the data in time-domain, while (b) and (c) is the data in frequency domain, (b) displays the frequency information in the full frequency range (0-81.4 Hz), (c) is a zoom in version of (b) which focuses on the half range of the frequency (0-40 Hz).

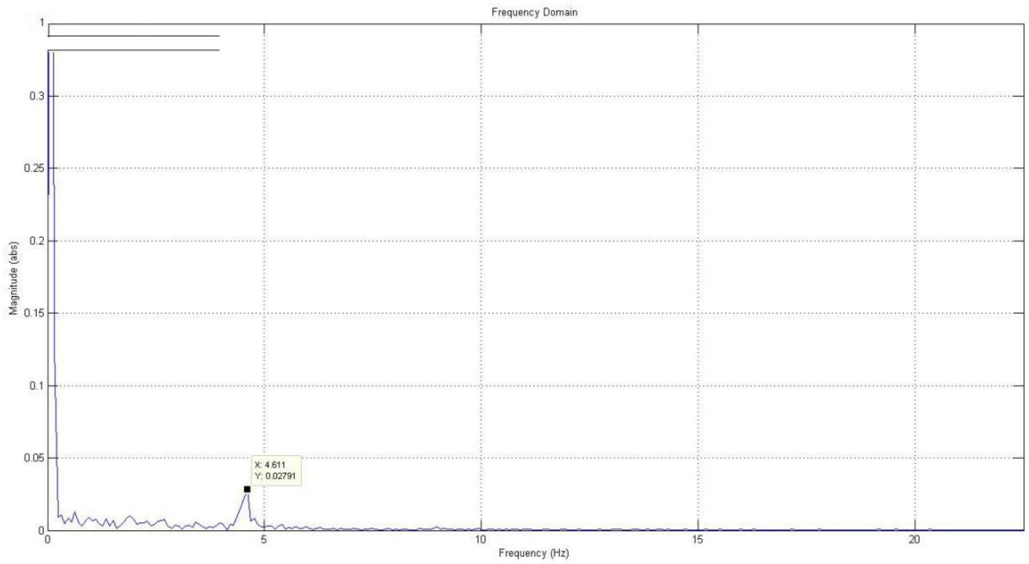
Figures 4-12 and 4-13, especially the frequency domain plots, show that significant frequency components exist around 5 Hz. But the waveform in the plot contains noise which interferes the decision making.

According to the smooth method introduced by (Smith 2013), the original signal could be multiplied by a smooth curve called Hamming window. It could suppress the noise around the significant frequency after the Fourier Transform which emphasises the real significant peak.

Compared with previous two figures, the smoothing process conveys more understandable peak information in Figure 4-14. According to Figure 4-14, the last peak is 5.007 Hz in (a), it means the highest significant frequency component of the signal in the dataset 1 is 5.007 Hz. Refer to the Nyquist theorem, if the sampling frequency for this signal is higher than 10.014Hz (5.007*2 = 10.014), the signal can be reconstructed without information loss of the main frequency component. Because both datasets in group one are collected in the same sitting posture pattern (very quick micro motion on thigh), their result is similar. In order to explore different patterns, this second stage of the experiment is carried out.



(a)



(b)

Figure 4-14 Smoothed Frequency Domain Plot.
(a) shows dataset 1 in group one, (b) shows dataset 2 group one. The least peak is 5.007 Hz in (a) and 4.611 Hz in (b).

4.3.2 The Frequency Domain Analysis on Second Data Group

In the second stage of the frequency experiment, the data collection and analysis procedure is similar as in the previous stage along with the curve smoothing process. The difference is that in this stage, data is collected with a higher sampling frequency (range from 190.6Hz to 198.9Hz), and the subject performs different human and chair contact pattern (see Table 4-4). The reason for the higher sampling frequency is to ensure more complete signal reconstruction, while the different sitting patter is to find suitable sampling frequency for the upcoming sitting activity experiment.

Based on the experiment result of the previous stage, the focus of this stage is to further explore the signal features of different levels of micro motion of different body parts. The correspondence between the dataset and the micro motion pattern is in table 4-5.

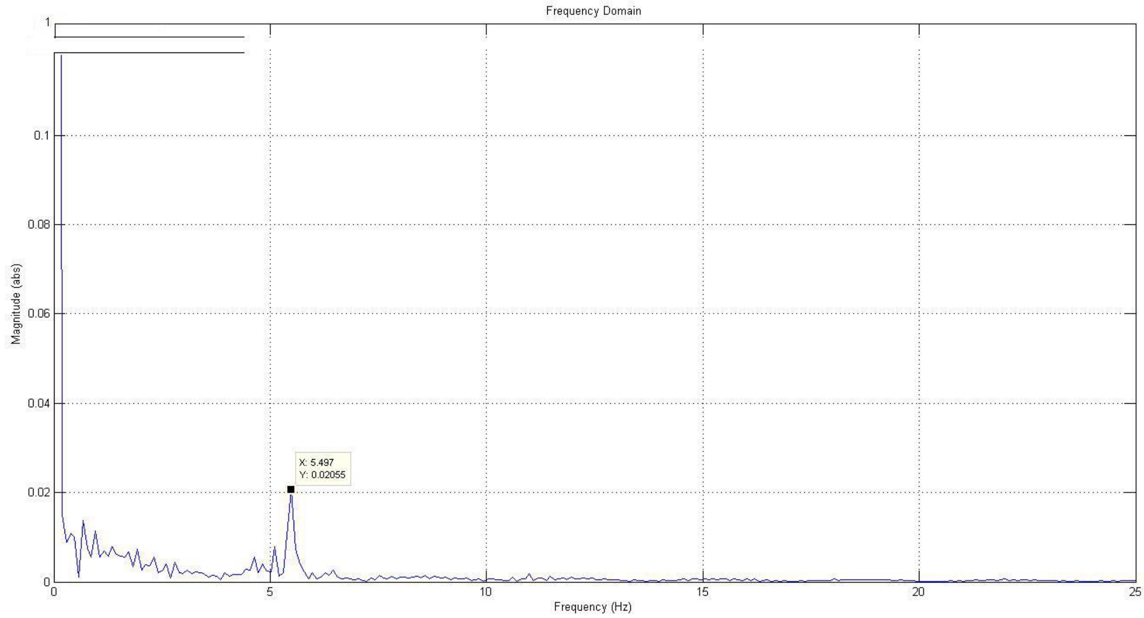
Because the major concern of this stage focuses on the frequency domain analysis, only frequency domain plots is shown in figure 4-15. Each of the plots in figure 4-15 represents the frequency domain feature of one micro motion pattern in sitting posture. Table 4-5 shows the signal frequency feature of the micro motion in human sitting posture, and its correlate Nyquist rate.

The Dataset Number 1, 2, 3, 4, 5 in Table 4-5 corresponds to (a), (b), (c), (d), (e) in Figure 4-15. The Body Segment column in Table 4-5 indicates which part of the body that generates the signal. The Micro Motion Level indicates the level of intensity that the body part contacts the chair surface. The Highest Frequency Component column indicates the last significant peak of a frequency domain plot.

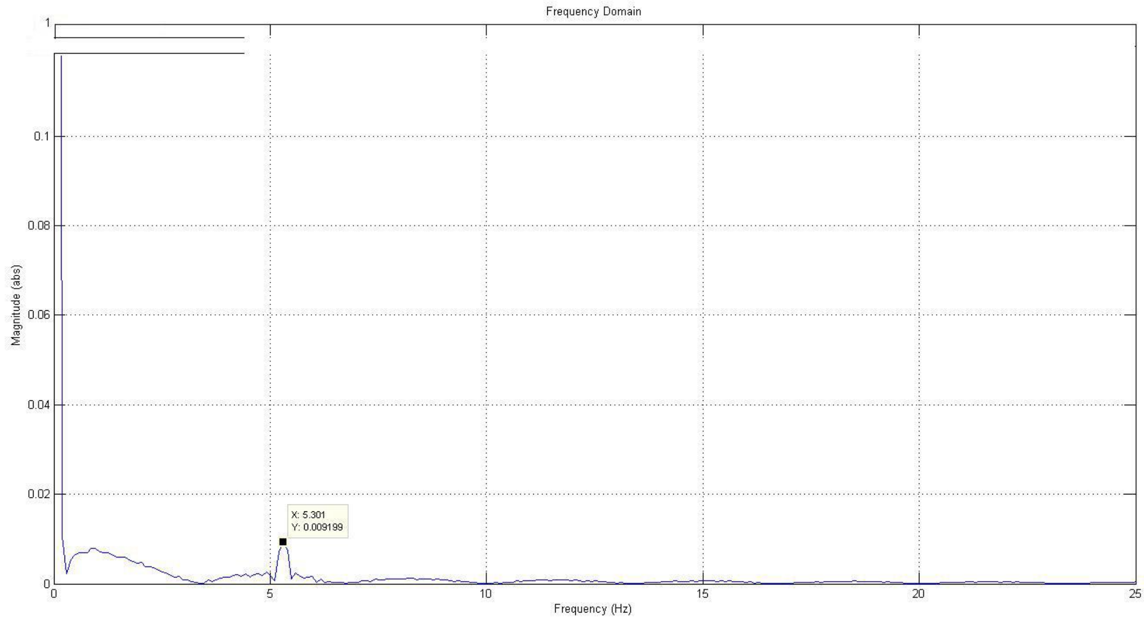
The value in the Nyquist Rate column is the most useful information. They are twice the value of the Highest Frequency Component column, which indicates the minimum sampling frequency that is capable to collect the corresponding micro motion intensity sitting data. For example, if the IntelliChair wants to collect the very quick shaking leg part data without losing any information, the minimum sampling frequency is 1.994 Hz (refer to dataset 1 row in Table 4-5). The experiment result is referred in section 4.3.3 as a proof of sampling frequency selection for activity data collection in the next stage of research.

Dataset Number	Body Segment	Micro Motion Level	Highest Frequency Component (Last Peak)	Nyquist Rate
1	Right Thigh	Very quick shaking.	5.497 Hz	10.994 Hz
2	Right Thigh	Quick shaking	5.301 Hz	10.602 Hz
3	Right Thigh	Normal Contact	2.715 Hz	5.430 Hz
4	Right Shoulder	Normal Contact	3.108 Hz	6.216 Hz
5	Right Hip	Normal Contact	2.978 Hz	5.956 Hz

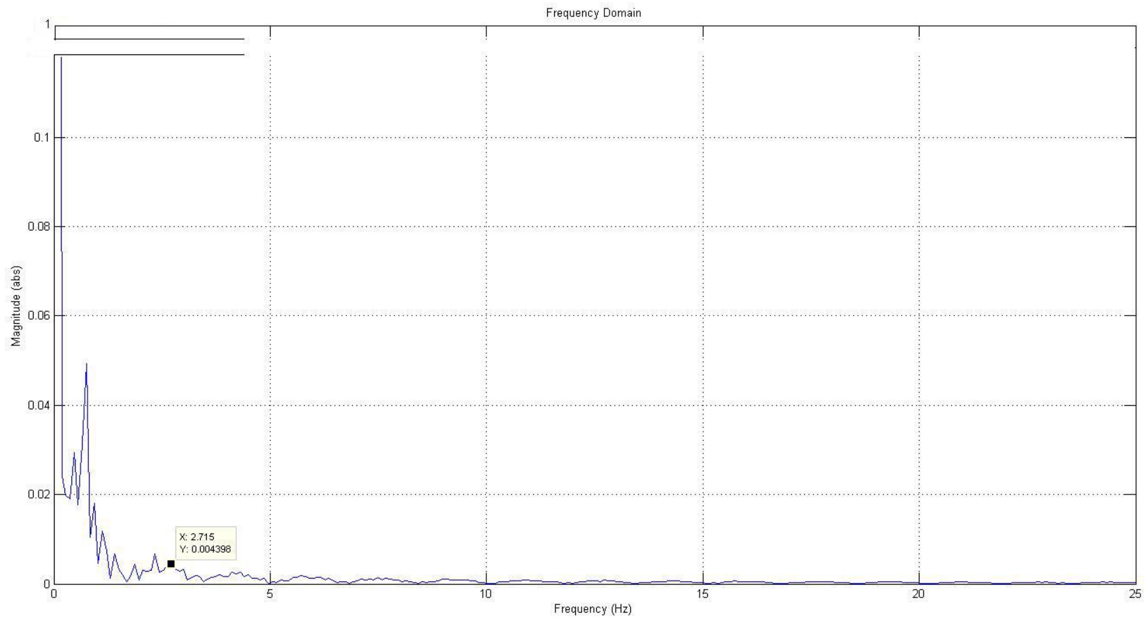
Table 4-5 The dataset and their highest frequency components feature.


Frequency Domain
X: 5.497
Y: 0.02055
Magnitude (abs)
Frequency (Hz)
1
0.1
0.08
0.06
0.04
0.02
0
0
5
10
15
20
25

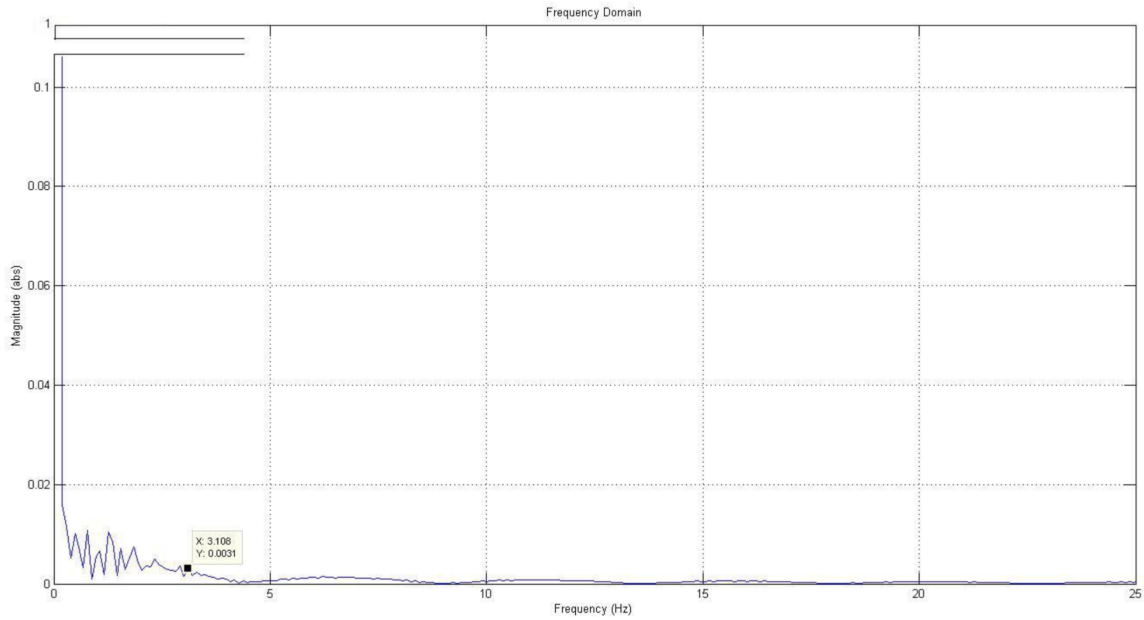
(a)


Frequency Domain
X: 5.301
Y: 0.009199
Magnitude (abs)
Frequency (Hz)
1
0.1
0.08
0.06
0.04
0.02
0
0
5
10
15
20
25

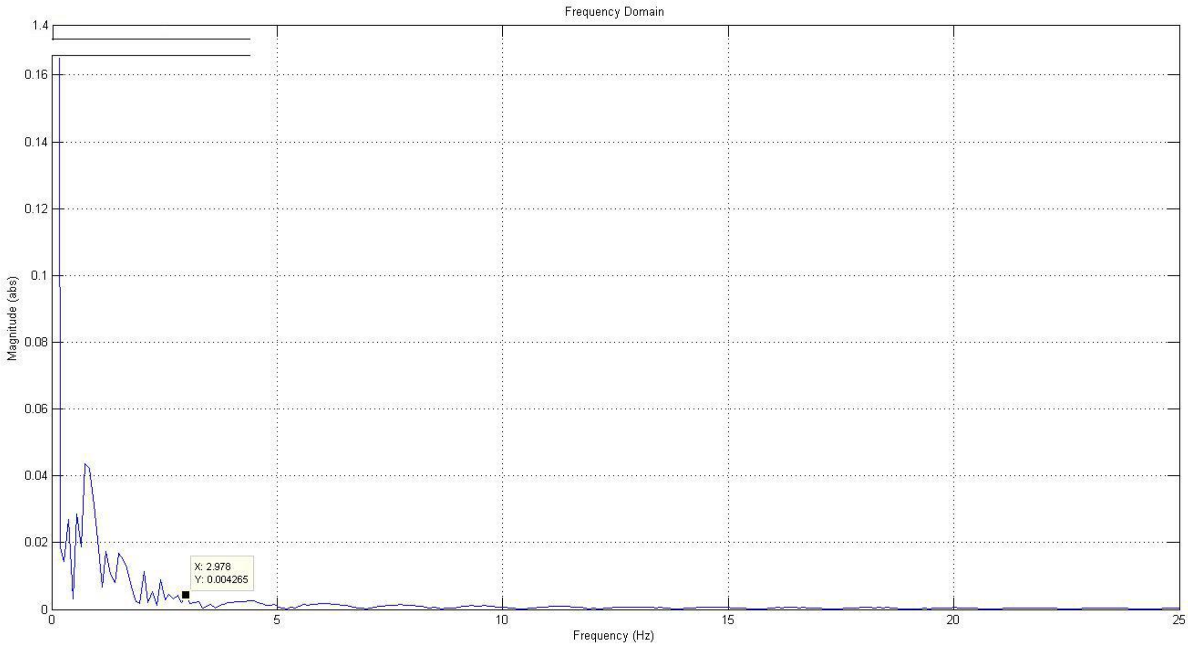
(b)


Frequency Domain
Magnitude (abs)
Frequency (Hz)
X: 2.715
Y: 0.004398

(c)


Frequency Domain
Magnitude (abs)
Frequency (Hz)
X: 3.108
Y: 0.0031

(d)



(e)

Figure 4-15 The Frequency Domain of the datasets in Data Group 2.

(a) represents dataset 1. (b) represents dataset 2. (c) represents dataset 3. (d) represents dataset 4. (e) represents dataset

4.3.3 Summary of Sampling Frequency Experiment

The aim of this experiment was to discovery the signal characteristic of the human sitting posture in order to determine the sampling frequency for IntelliChair system.

According the result in table 4-5 the signal of normal sitting posture micro motion changing frequency is 3.1 Hz maximum (2.7 Hz, 3.1 Hz and 2.98 Hz in table 4-5), while the quick micro motion changing frequency is around 5.5 Hz maximum (5.5 Hz and 5.3 Hz in table 4-5). Based on this result and the Nyquist Theorem, the suitable sampling frequencies are:

- 6.2 Hz if the system is monitoring normal sitting activity (e.g. reading, relaxing).
- 11 Hz if the system is monitoring anomalous sitting activity (e.g. stroke, nervous, or stress detection).

The result of this experiment not only answers the third research question (objective 1c) (What is the signal characteristic of the pressure sensing information?) in the beginning of chapter 3, but also provides a baseline for researchers that aim to build systems for anomalous activity detection purposes. IntelliChair aims to monitor the normal sitting activities (e.g. relaxing, watching video, etc.), so its sampling frequency should be 6.2 Hz or higher. But limited by the hardware of Raspberry Pi and ADC Pi v2 board, the overall sampling frequency for IntelliChair can only maintain 6 Hz in the activity experiment. Consider the minor difference between 6 and 6.2, six Hz is acceptable and therefore, is the sampling frequency used for the activity data collection process that is described in chapter 5 in this thesis.

4.4 Experiment of Sitting Posture Classification

The aim of this experiment is to train two classifiers (one for spine postures and the other one for leg postures) that enable the system to detect different sitting postures which answers the research question 4 (objective 2): How to classify the data in order to detect the sitting posture?. In order to train the classifiers, the pressure information based posture dataset with specific posture labels is collected in this experiment through the IntelliChair system as well.

The content of this section includes the details of the sitting posture classification experiment from training data collection through the estimation of the Support Vector Machine (SVM) classification algorithms and evaluation of the result.

The structure of this section is as follow:

- Section 4.4.1: Describes the posture data collection procedure for this experiment.
- Section 4.4.2: Explore and explain the posture data statistically.
- Section 4.4.3: Gives a short overview of classification algorithms explains the reason why SVM is selected.
- Section 4.4.4: Parameter estimation of SVM classifier.
- Section 4.4.5: Evaluation of SVM classifier.

There are 10 participants involved in this experiment; they are not only involved with the posture data collection but also the activity data collection. Their basic information is in table 4-6. The decision of ten participants is based on the research from Mutlu (20 subjects) (Mutlu et al. 2007) and Cheng (5 subjects) (Cheng et al. 2013). Considering the subjects in this experiment also needs to participant the four hours activity experiment (see chapter 5), the number of 10 subjects is decide to meet the acceptable subject coverage.

Subject ID	Gender	Weight	Height
Subject 01	Male	108 KG	193 cm
Subject 02	Male	62 KG	167 cm
Subject 03	Male	95 KG	190 cm
Subject 04	Female	67 KG	170 cm
Subject 05	Male	78 KG	173 cm
Subject 06	Male	80 KG	175 cm
Subject 07	Male	72 KG	167 cm
Subject 08	Female	71 KG	175 cm
Subject 09	Male	92 KG	175 cm
Subject 10	Male	93 KG	175 cm

Table 4-6 The Subject ID and their personal characteristics.

4.4.1 Data Collection for Posture Classification

There are two stages for classification, the training stage and the classification stage. The training stage requires a given set of labelled training samples to build a classifier, and then the trained classifier can be used to classify unknown samples into certain pre-defined classes. The aim of the experiment is to establish a training dataset for training stage of posture classification. This aim requires the data collection system to label each sample in the data record with the correct posture label. Furthermore, in order to evaluate the classification performance when the classifier is dealing with postures from different subjects, the data collection is repeated for different participants.

In order to tackle the requirement for data collection, a Python based GUI program is developed to collect both pressure data and automatically generated corresponding posture labels. The sitting postures and their correlated labels are listed in Table 3-1 in section 3.3.4.

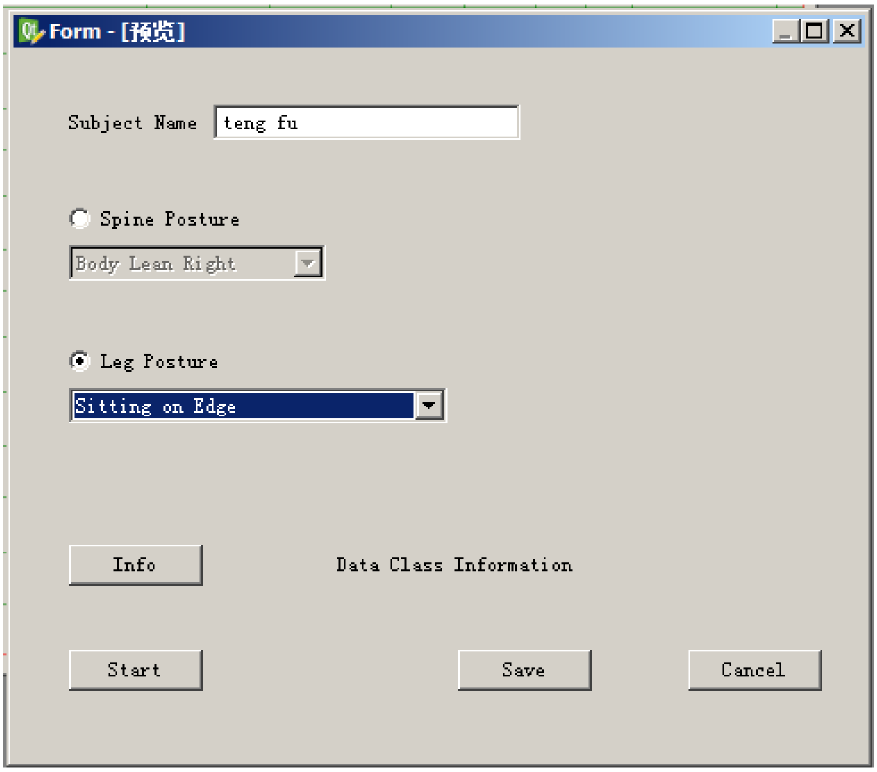


Figure 4-16 The GUI for posture data collection.

According to the discussion in section 3.3.4, the16 postures are divided into two sets. The first set is the Spine Posture group, and another set is the Leg Posture group. It is shown as two different selection radio buttons; clicking on one of them will disable the other one. The two reasons for the separation is discussed in section 3.3.4, the first reason is to make the participants more focus when performing certain posture, second reason is to eliminate the cross interfere between spine posture and leg postures.

According to the discussion, the separate design has two benefits:

- This separation is to make sure the participants concentrate on the correct pressure sensing area in order to maximize the quality of the data. For example, posture 3 (Body Leaning Back) and posture 14 (Leaning Back) are the same posture, but the difference is that posture 3 requires the participants focus on their back to make contact with the vertical surface of the Chair, while posture 14 requires the participants to focus on the pressure which is generated by their thigh.
- Another benefit from this separation is to improve the sampling frequency of IntelliChair system which reduces the time for the data collection. Because there are four sensors deployed on the vertical surface of the chair and the other four sensors are deployed on the horizontal surface of the chair. When collecting the spine part or the leg part posture data, only the data from the four sensors on the vertical surface or horizontal surface is needed to be recorded. This will allow the sampling frequency of the IntelliChair system increase from 12Hz to 25Hz, which decreased the collection time from more than 20 seconds to around 10 seconds for a single posture collection process. This helps the participants to have better experience in the experiment.

Along with the GUI, data collection procedures are as follow:

1. Participants input their names in the Subject Name text field.
2. Experimenter selects the specific posture; the participants will perform the posture as required.
3. Experimenter clicks the *start* button to start the pressure data collection process.
4. The system will collect 250 samples for each posture; the whole collection process takes 10 seconds for a single posture collection process. Total 16 postures will take 3 or 4 minutes because sometimes participants need some time to adjust their posture.
5. When the collection process is done, the experimenter click the *save* button, the system will automatically save the data following the naming rules:

- The raw data file:

  <subject name>_<posture name>_raw_data.

  (e.g. Teng_Fu_leg_Crossing_Right_Leg_on_Left_Leg_raw_data)

- The target data file:

  <subject name>_<posture name>_target_data_<label number>

  (e.g. Teng_Fu_leg_Crossing_Left_Leg_on_Right_Leg_target_data_7)

6. The collecting process will repeat across ten participants.

The raw data file contains the 4 channels of pressure information, each channel represent the data from one sensor. The target data file contains the label number which matches to the sample numbers in raw data file. The uniform naming rules not only help the experiment process, but also the following data analysis process.

After this data collection process, the first stage of the Posture classification experiment is accomplished with training data that contains 16 postures from 10 participants. There are 250 samples for each posture from each participant, so there are 2500 samples for each posture, and total 40000 samples of all postures.

4.4.2 Posture Dataset Explanation and Statistics

The raw data recorded by IntelliChair pressure sensing system are firstly analysed statistically in this section. The purpose of this is to evaluate the distribution of the pressure data (mean, standard deviation, etc.) for further sitting posture classification purpose.

The variables involved in sitting posture classification were determined and they are presented as follow.

- Spine part: CH_1, CH_2, CH_3 and CH_4.
- Leg part: CH_5, CH_6, CH_7 and CH_8.

There are eight Pressure values (*CH* stands for channel), each of the value represents the voltage measurement from one sensor. Because each posture consists of either spine or leg part but not both, so each posture is described by 4 pressure values:

$$P_{spine}^{s} = \left\{CH_1^s, CH_2^s, CH_3^s, CH_4^s\right\}, s \in \left\{s \in \mathbb{N} : 1 \leqslant s \leqslant 4\right\}$$

$$P_{Leg}^{l} = \left\{CH_5^l, CH_6^l, CH_7^l, CH_8^l\right\}, l \in \left\{l \in \mathbb{N} : 5 \leqslant l \leqslant 16\right\}$$

$$CH_i^s, CH_i^l \in \left\{CH_i \in \mathbb{R} : 0 < CH_i < 5\right\}$$

CH_i represents the values of sensor i, it is a real number between 0 to 5 because of the signal conditional circuit and it cannot be equal to either 0 nor 5. The s and l is the index of posture, they are integer numbers and 1 to 4 means 4 spine posture, 5 to 16 means 12 leg posture.

The amount of pressure data consists in a total of

$$\left[\left(f^{frequency} \cdot t \cdot n^{sensosrs}\right) \cdot m^{postures}\right] \cdot n^{participants}$$

Where $f^{frequency} \approx$ 25 Hz is the average sampling frequency of the IntelliChair pressure sensing system, $t =$ 10 Seconds means the length of time for each posture data collection and $n^{sensosrs} =$ 8 means there are 8 sensors. $m^{postures} =$ 16 are the number of postures and $n^{participants} =$ 10 is the number of participants. The overall number is around 320000 because of the sampling frequency variation.

The purpose of the statistical analysis is to explore the feature of each posture pressure dataset according to its statistic parameters (mean, standard deviation, etc.). SPSS software is utilized for the pressure data statistical analysis. SPSS software also provides the box plot for statistic parameters detail for each posture data that is collect in section 4.4.1.

The statistical analysis result will be reported in section 4.4.2. The averaged pressure distribution is estimated in table 4-7 . There are 16 tables in table 4-7 from (a) to (p), and they are correlated to posture 1 to 16. Tables show the statistical parameters including Mean, Standard Deviation (STD), Standard Error (STE) and Confidence Interval (CI) high end and confidence interval low end (confidence interval: 95%).

There are 16 figures from (a) to (p) in Figure 4-17, which represent the statistic parameters reported in Table 4-7 (a) to (p). Those figures show how the data are spread around the average values, the high and low inter-quartile values and the covering range. The y axis of the box plots represents the voltage (*V*) measurement of the FSR sensor which indicates the strength of force applied.

Figure 4-17 The Box Plot for pressure data for 16 postures.
The y axis represents the voltage (*V*) measurement of the FSR sensor which indicates the strength of force applied (Overleaf, with 16 sub-figures from (a) to (p), correlate with 16 postures from posture 1 to posture 16).

Table 4-7 The Statistic parameters obtained by SPSS (Overleaf, with 16 sub-tables from (a) to (p), correlate with 16 postures from posture 1 to posture 16).

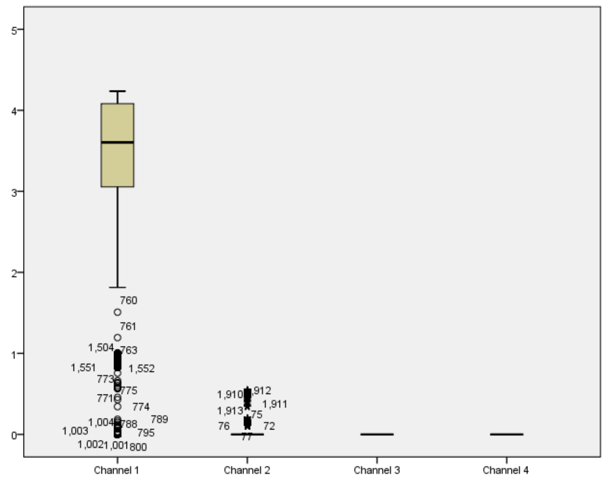


Figure 4-17 (a) Spine Posture 1: Body Lean Right

Posture 1	Mean	STD	STE	CI(Low)	CI(High)
Channel 1	3.049	1.417	0.030	2.990	3.108
Channel 2	0.070	0.157	0.003	0.063	0.0767
Channel 3	Constant 0	Constant 0	Constant 0	N/A	N/A
Channel 4	Constant 0	Constant 0	Constant 0	N/A	N/A

Table 4-7 (a) Spine Posture 1: Body Lean Right

Posture 1: The significant feature for this posture 1 is the value from channel 1 because it is much higher than the rest of the three channels. The value from channel 2 is close to null, and value from channel 3 and 4 is null.

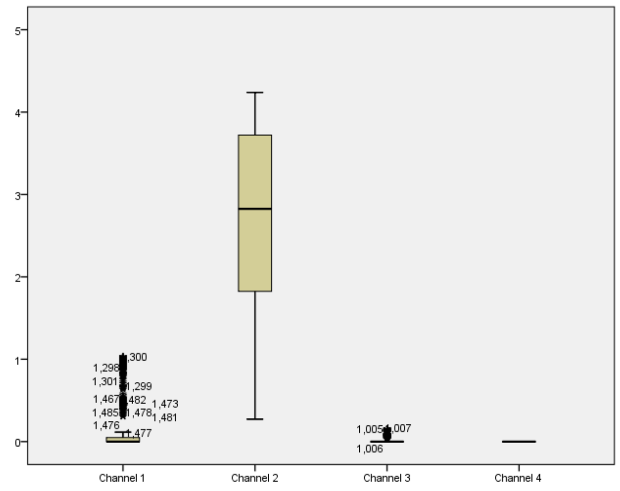


Figure 4-17 (b) Spine Posture 2: Body Lean Left

Posture 2	Mean	STD	STE	CI(Low)	CI(High)
Channel 1	0.17	0.328	0.007	0.16	0.18
Channel 2	2.812	1.148	0.024	2.764	2.859
Channel 3	0.01	0.030	0.001	0.01	0.01
Channel 4	Constant 0	Constant 0	Constant 0	N/A	N/A

Table 4-7 (b) Spine Posture 2: Body Lean Left

Posture 2: Channel 2 is the significant factor for posture 2, while the value from channel 1 and 3 are close to null, and value from channel 4 is 0.

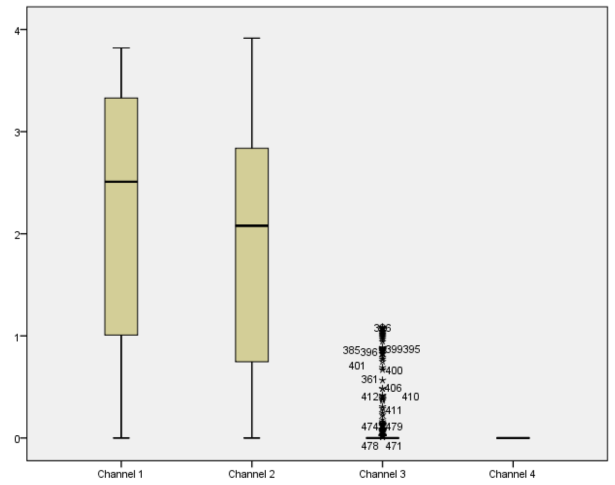


Figure 4-17 (c) Spine Posture 3: Body Leaning Back

Posture 3	Mean	STD	STE	CI(Low)	CI(High)
Channel 1	2.228	1.244	0.026	2.177	2.280
Channel 2	1.946	1.223	0.026	1.895	1.996
Channel 3	0.02	0.127	0.003	0.02	0.03
Channel 4	Constant 0	Constant 0	Constant 0	N/A	N/A

Table 4-7 (c) Spine Posture 3: Body Leaning Back

Posture 3: Both values from channel 1 and 2 are the factor for this posture, and their mean values and STD are fairly similar. The value from channel 3 is close to 0 while value from channel 4 is 0 as well.

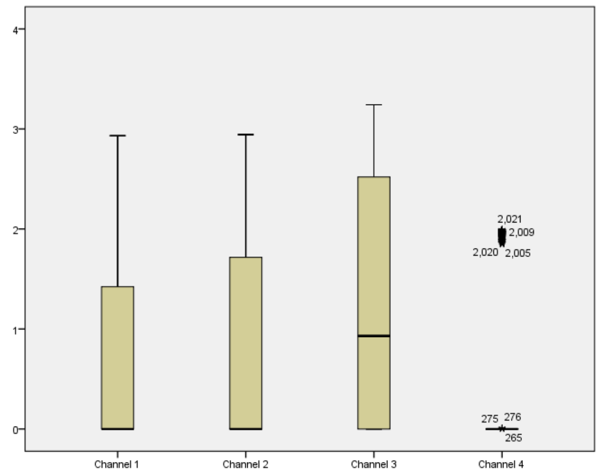


Figure 4-17 (d) Spine Posture 4: Body Crouch

Posture 4	Mean	STD	STE	CI(Low)	CI(High)
Channel 1	0.721	1.015	0.021	0.680	0.764
Channel 2	0.823	1.150	0.024	0.776	0.871
Channel 3	1.199	1.125	0.024	1.153	1.246
Channel 4	0.22	0.611	0.013	0.19	0.24

Table 4-7 (d) Spine Posture 4: Body Crouch

Posture 4: The values from channel 1 and 2 are close, and the value from channel 3 is the main factor, because its average value is roughly 40% higher than channel 1 and 2.

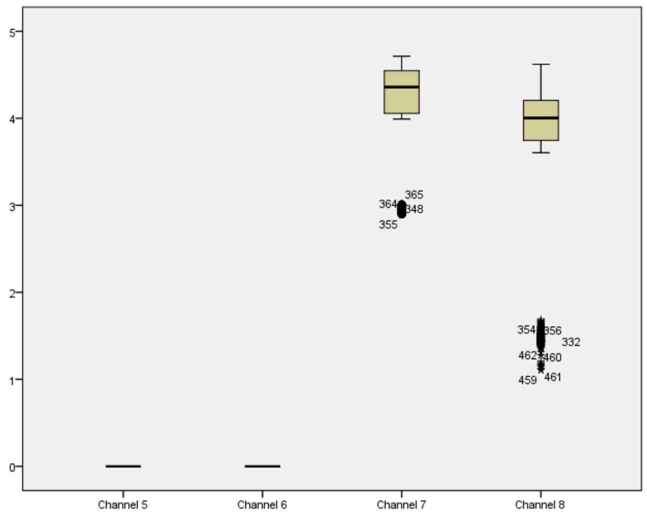


Figure 4-17(e) Sitting Posture 5: Sitting on Edge

Posture 5	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	Constant 0	Constant 0	Constant 0	N/A	N/A
Channel 6	Constant 0	Constant 0	Constant 0	N/A	N/A
Channel 7	4.224	0.506	0.011	4.203	4.245
Channel 8	3.797	0.880	0.019	3.761	3.834

Table 4-7 (e) Sitting Posture 5: Sitting on Edge

Posture 5: The main factor are the values from channels 7 and 8. The values from channel 5 and 6 are 0.

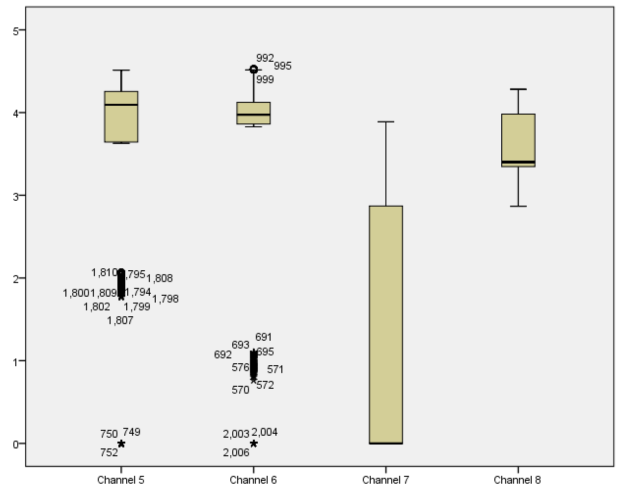


Figure 4-17(f) Sitting Posture 6: Crossing Right Leg on Left Leg

Posture 6	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	3.403	1.402	0.029	3.345	3.461
Channel 6	3.293	1.533	0.032	3.230	3.357
Channel 7	1.112	1.524	0.032	1.049	1.175
Channel 8	3.591	0.422	0.009	3.573	3.608

Table 4-7 (f) Sitting Posture 6: Crossing Right Leg on Left Leg

Posture 6: The mean values from channel 5, 6 and 8 are close and they are much higher than the value from channel 7.

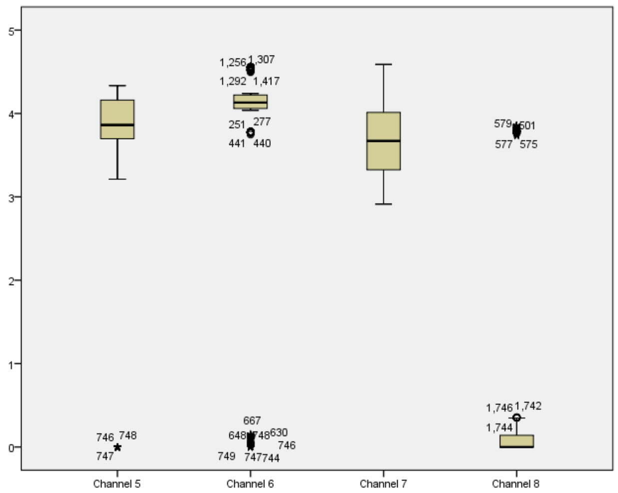


Figure 4-17 (g) Sitting Posture 7: Crossing Left Leg on Right leg

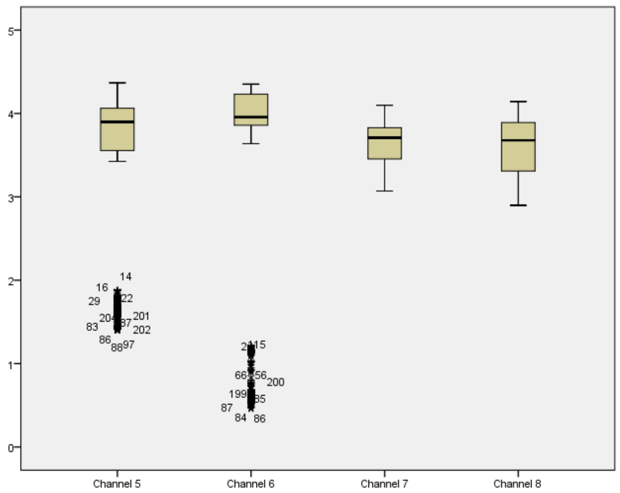


Figure 4-17 (h) Sitting Posture 8: Sitting Forward

Posture 7	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	3.487	1.264	0.027	3.435	3.539
Channel 6	3.728	1.325	0.028	3.674	3.783
Channel 7	3.687	0.485	0.010	3.667	3.707
Channel 8	0.46	1.178	0.025	0.41	0.51

Table 4-7 (g) Sitting Posture 7: Crossing Left Leg on Right leg

Posture 8	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	3.682	0.779	0.016	3.650	3.714
Channel 6	3.676	1.068	0.022	3.632	3.720
Channel 7	3.620	0.319	0.007	3.620	3.647
Channel 8	3.577	0.363	0.008	3.562	3.592

Table 4-7 (h) Sitting Posture 8: Sitting Forward

Posture 7: The mean values from channels 5, 6 and 7 are close and they are much higher than the value from channel 8.

Posture 8: The values from all four channels are close and around 3.6.

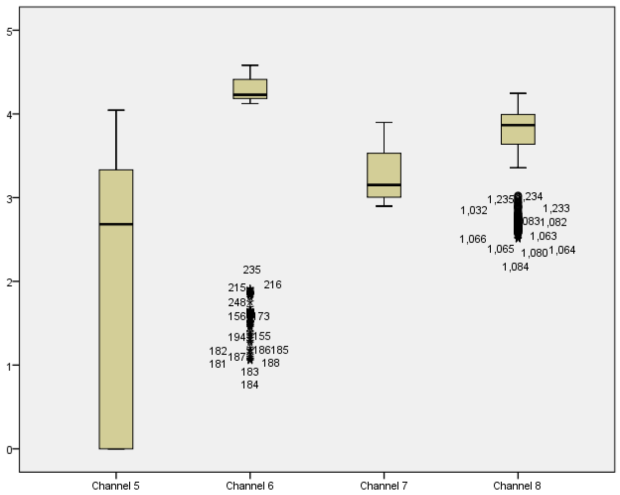


Figure 4-17 (i) Sitting Posture 9: Sitting Forward Left

Posture 9	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	2.120	1.555	0.033	2.060	2.180
Channel 6	4.012	0.863	0.018	3.977	4.048
Channel 7	3.289	0.309	0.007	3.276	3.302
Channel 8	3.712	0.424	0.009	3.694	3.729

Table 4-7 (i) Sitting Posture 9: Sitting Forward Left

Posture 9: The values from channel 6, 7 and 8 are relatively higher than channel 5. The mean value of channel 6 reaches 4 V and it tops channel 7 and 8.

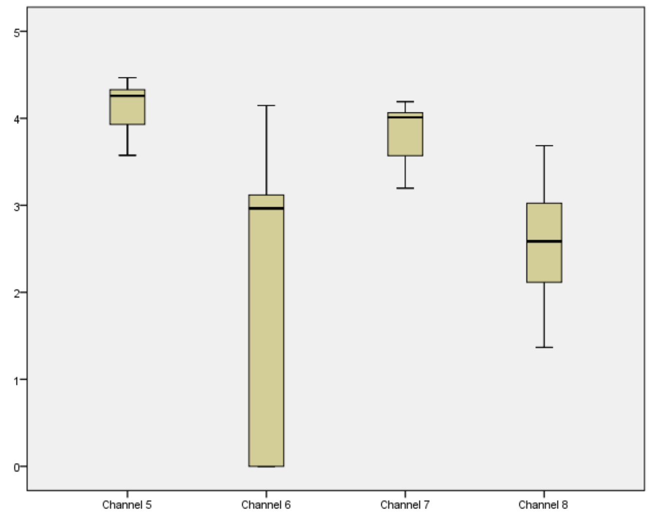


Figure 4-17 (j) Sitting Posture 10: Sitting Forward Right

Posture 10	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	4.151	0.251	0.005	4.140	4.161
Channel 6	2.090	1.564	0.033	2.030	2.160
Channel 7	3.845	0.322	0.007	3.832	3.858
Channel 8	2.622	0.585	0.123	2.598	2.646

Table 4-7 (j) Sitting Posture 10: Sitting Forward Right

Posture 10: The mean value from channels 5 and 7 are the highest, channel 6 is the lowest (but the medium of channel 8 is less than medium of channel 6). The mean value of channel 5 reaches 4.1 V which is higher than channel 7 and 8.

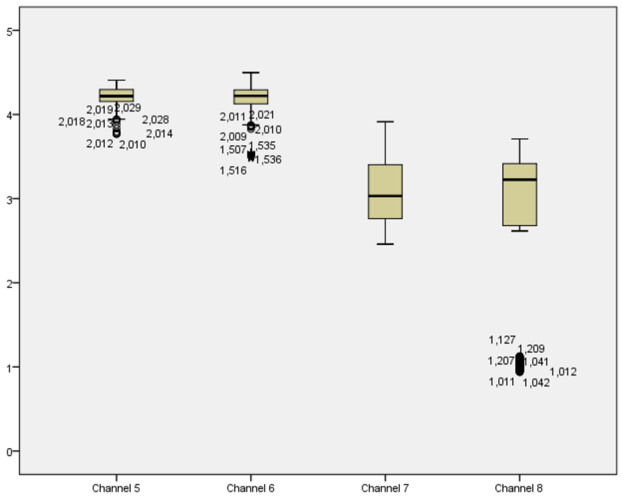


Figure 4-17 (k) Sitting Posture 11: Sitting Upright

(a) Posture 11	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	4.212	0.138	0.003	4.206	4.218
Channel 6	4.161	0.264	0.006	4.150	4.172
Channel 7	3.123	0.472	0.010	3.104	3.143
Channel 8	2.934	0.764	0.016	2.903	2.966

Table 4-7 (k) Sitting Posture 11: Sitting Upright

Posture 11: The values from channel 5 and 6 are close and their mean values are more than 4.1 V, and they are 29% higher compare with mean values of channel 7 and 8. The values from 7 and 8 are also close.

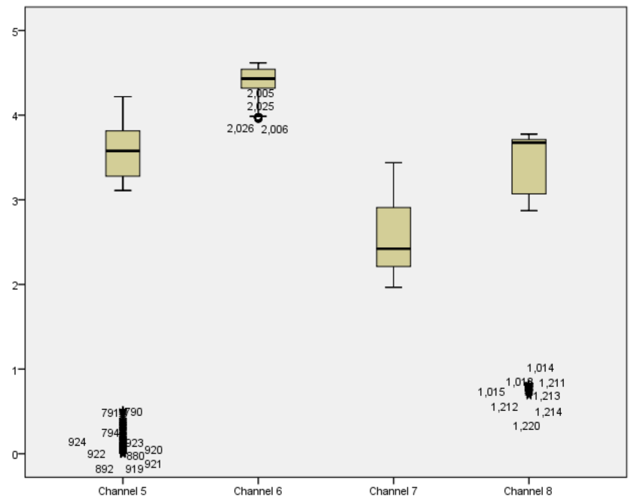


Figure 4-17 (l) Sitting Posture 12: Sitting Upright Left

Posture 12	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	3.230	1.130	0.024	3.183	3.277
Channel 6	4.368	0.219	0.005	4.559	4.378
Channel 7	2.580	0.432	0.010	2.561	2.597
Channel 8	3.164	0.899	0.0190	3.127	3.201

Table 4-7 (l) Sitting Posture 12: Sitting Upright Left

Posture 12: The mean value of channel 6 is 4.368 V, top of four channels. Mean value from channel 6 is 26% higher than channel 5 while mean value from channel 8 is 18% higher than channel 7.

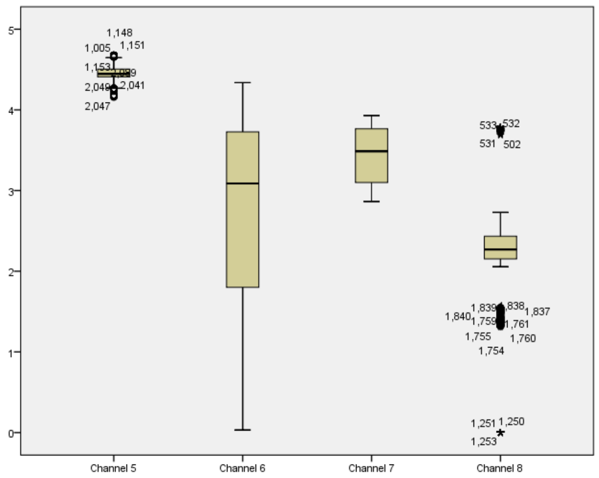


Figure 4-17 (m) Sitting Posture 13: Sitting Upright Right

Posture 13	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	4.430	0.135	0.003	4.424	4.435
Channel 6	2.740	1.234	0.026	2.690	2.791
Channel 7	3.444	0.369	0.008	3.428	3.459
Channel 8	2.146	0.947	0.020	2.107	2.185

Table 4-7 (m) Sitting Posture 13: Sitting Upright Right

Posture 13: The mean value of channel 5 reaches 4.43 V, top of four channels. Mean value from channel 5 is 38% higher than channel 6 while mean value from channel 7 is 37% higher than channel 8.

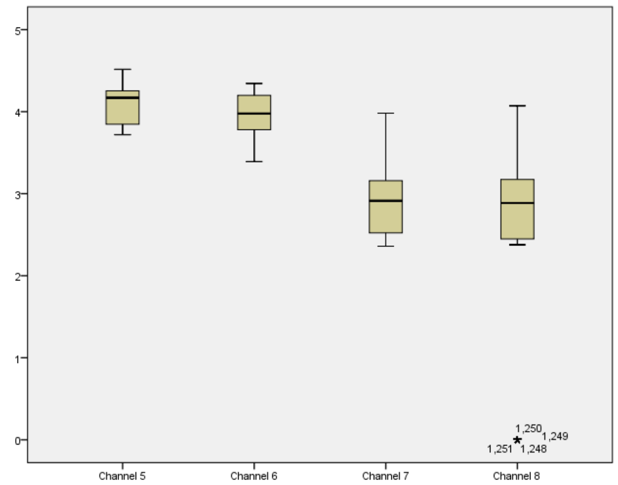


Figure 4-17 (n) Sitting Posture 14: Leaning Back

Posture 14	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	4.109	0.241	0.005	4.099	4.119
Channel 6	3.946	0.267	0.006	3.935	3.957
Channel 7	2.968	0.461	0.009	2.949	2.987
Channel 8	2.633	1.011	0.021	2.591	2.675

Table 4-7 (n) Sitting Posture 14: Leaning Back

Posture 14: The values from channel 5 and 6 are close as are channel 7 and 8. But they are 33% higher compared with mean values of channel 7 and 8. And their mean values are around 4 V.

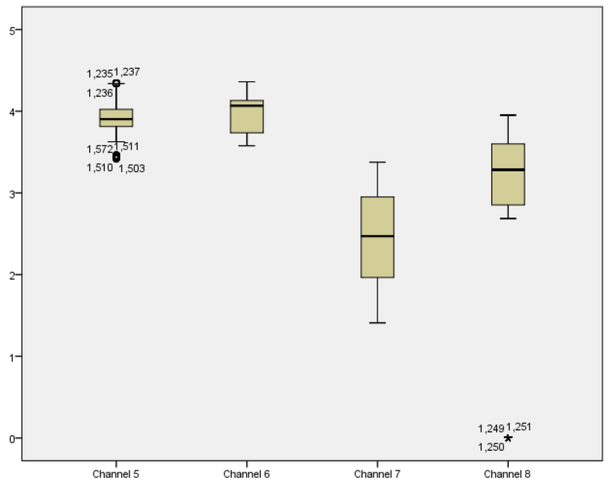


Figure 4-17 (o) Sitting Posture 15: Leaning Back Left

Posture 15	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	3.907	0.249	0.005	3.897	3.917
Channel 6	3.996	0.247	0.005	3.986	4.006
Channel 7	2.413	0.616	0.013	2.387	2.438
Channel 8	2.950	1.108	0.023	2.904	2.995

Table 4-7 (o) Sitting Posture 15: Leaning Back Left

Posture 15: The mean value of channel 6 is slightly higher than channel 6, and they are more than 3.9. The value from channel 8 is 18% higher than channel 7. The mean values of channel 5 and 6 are overall higher than channel 7 and 8.

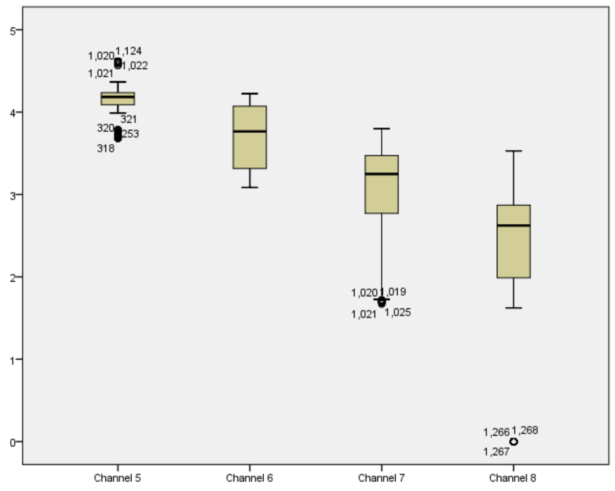


Figure 4-17 (p) Sitting Posture 16: Leaning Back Right

Posture 16	Mean	STD	STE	CI(Low)	CI(High)
Channel 5	4.182	0.225	0.005	4.172	4.191
Channel 6	3.721	0.370	0.008	3.706	3.737
Channel 7	3.019	0.667	0.014	2.992	3.047
Channel 8	2.292	0.953	0.020	2.253	2.331

Table 4-7 (p) Sitting Posture 16: Leaning Back Right

Posture 16: The mean value of channel 5 and 6 are overall higher than channel 7 and 8. The mean value of channel 7 is 24% higher channel 8, while channel 5 is 10% higher than channel 6.

Through the statistical analysis on the pressure, the features for each posture can be shown and they are discussed in spine group and leg group. The spine posture group (from (a) to (d) in figure 4-17) has channel 1 and 2 for clear indication, while channel 4 has no effect on posture indication.

Spine Postures	
Posture	Comment
1 (body lean right)	They can be detected by the values from channel 1 and channel 2, since the difference between them is significant.
2 (body lean left)	
3 (body leaning back)	
4 (body crouch)	Values from channel 3 are the main indicator

Table 4-8 Feature discussion table for Spine Posture.

For the sensors on the horizontal surface that detects Leg posture group (from (d) to (p) in Figure 4-17), the channel 5 and channel 7 sensors represent the pressure data from right hip and right thigh, while channel 6 and channel 8 represents the pressure data from left hip and left thigh (see the sensor placement Figure 4-7 in section 4.2.1 for easy understanding). According to this sensor placement, the postures with mirror symmetry features are reflected in the statistic data. The four pairs of postures listed below show that the values from channel 5 and 7, and the values from channel 6 and 8 are relatively symmetrical.

Leg Postures	
Posture	Comment
6 (crossing right leg on left leg) 7 (crossing left leg on right leg)	Both channel 5 and channel 6 are high values, and the difference between channel 7 and channel 8 is the main feature where posture 6 is the opposite of the posture 7.
9 (sitting forward left) 10 (sitting forward right)	As expected, the figures are near mirror image for posture 9 and 10. Channels 5 and 7 are similar to each other, 6 and 8 are similar to each other.
12 (sitting upright left) 13 (sitting upright right)	Similar to the last pair, but the mean parameters of the value from channel 5 and 6 are higher than last pair.
15 (leaning back left) 16 (leaning back right).	The different between the values from channel 5 and channel 6 are much less than previous three pairs of postures.
5 (sitting on edge)	Channel 5 and 6 are 0 while channels 7 and 8 are high and similar to each other.
8 (sitting forward)	The mean values from channels are very similar to each other.
11 (sitting upright)	The mean values from channel 5 and 6 are very similar and higher than the mean values from channels 7 and 8 which are close to each other.
14 (leaning back)	Very similar to posture 11, but all values are slightly lower.

Table 4-9 Feature discussion table for Leg Posture.

The ranges of values in some posture are wide, and this wide range of values would cause wide confidence interval as showed in Figure 4-17. This is likely due to the influenced by individual body shape differences of the participants and their sitting habits.

Through the discussion, the conclusions can be stated as:

- The sitting posture classification is possible through the voltage values from sensors at critical position.
- All spine postures have significant pressure patterns factor for posture detection.
- Leg postures have 12 classes, 8 of them are mirror symmetry postures. Some of the classes have similar statistic features (posture 11 and 14), but the difference in values between them should allow for classification.

4.4.3 Overview of Classification Algorithms

The task of classification process in Machine learning area is to determine which pre-defined category an input sample or instance belongs to. Classification is a supervised learning method in machine learning, because this type of learning process relies on pre-defined categories which are provided with the actual outcomes for each of the samples in the training dataset.

In this experiment, The Support Vector Machine classification algorithms are utilized to fulfil the sitting posture classification task. Support Vector Machine (SVM) is from the research area of statistical learning theory. It is based on the theory of maximum margin hyperplane (Witten and Frank 2005). The hyperplane is a boundary that could determine different classes, and the maximum margin means the sum of the distance from both classes to the boundary are the largest. In the early stage, SVM is used for linear classification. Through the implementation of non-linear transformation and Kernel Trick (a method that maps low dimensional data into high dimensional data, therefore, separates the crowd classes. And this process makes the classes are able to be classified.), SVM is able to tackle the non-linear classification problem. The Radial Basis Function (RBF) kernel function is a popular solution for non-linear classification (Hsu and Lin 2002, Hsu, Chang and Lin 2010, Gunn 1998).

The reason for this choice of classification algorithm is not only because SVM is currently a popular classification method in a range of machine learning area (text classification, hand writing recognition, etc.)(Hsu, Chang and Lin 2010). Compared with other classification algorithms, such as the instance-based k-Nearest Neighbours classifier and the rule based classification decision tree, the advantage of SVM is because it avoids large amount of computation in the decision making process. Its decision function computation complicity depends on the number of Support vectors, instead of the dimension of the sample or the number of samples in the training data (see section 2.1.2).

4.4.4 Support Vector Classifier Estimation

The content of this section is about the classifier estimation for Support Vector Machine (SVM) based classifier. It follows the machine learning process of model estimation:

1. Data Pre-processing
2. Parameter estimation for SVM based classifier
3. Model evaluation

The Grid Search and Cross Validation are two important processes in data pre-processing stage. The Grid Search method is a model selection method for parameter search in machine learning area, because the parameter sets are unknown before a model is estimated. The solution is to define a parameter space and implement the hyperparameter into the model one by one and find out the best parameter set.

In machine learning, the split between training data and test data is a common strategy to maximise the usage of the dataset at hand. The dataset is split into two parts: training dataset and test dataset. The training dataset is used to train the model, and the test dataset aims to evaluate the model after training stage. The purpose of this split is to avoid a situation that uses the ‘known’ dataset to test the trained model, because this cannot reflect the true performance of a model. The prediction accuracy from an ‘unknown’ dataset is more meaningful. The common split between training dataset and test dataset is 2:1, where the training dataset is two thirds of the whole dataset and the test dataset is one third. This is also the split used for the whole posture data in this experiment.

This experiment uses a stratified splitting method, which creates splits by preserving the same proportion for each target class as in the complete set (i.e. there are 16 postures in the whole dataset, after the split, the proportion is still 1/16 in both training and test dataset for each class). The data are further shuffled to prevent instance with the same label being contiguous which is known to make the cross-validation result more reliable (Witten and Frank 2005, pp. 126).

In the training stage, especially in the grid search procedure, there is a risk that the knowledge about the test set is leaked to the model which could make the model perform optimally in the testing stage; this is called over-fitting. In order to prevent over-fitting problems in the training stage, the concept of a validation dataset is proposed. The validation is also a part of the whole dataset, but it is different from both training dataset and test dataset. It is used after the training is done, and validation dataset is used to perform a pre-evaluation process. When the result seems to be successful, the test dataset is used for final evaluation.

However, splitting the whole dataset into three sets reduces the number of samples that could be used for the training process, and the result depends on the choice of a particular training and validation set. The Cross Validation (CV) method is a procedure to address this problem. This procedure splits the training dataset into number of folds (K-fold), and then uses K-1 folds as training data, and leaves one fold as a validation dataset; this process repeats K times across the whole training dataset. The performance measured by K-fold CV is the average of the model performance through the iterations. This procedure maximizes the training dataset usage and achieves an average performance before entering into the test stage.

For evaluation purposes, Accuracy is the main metric for classification performance. In addition to Accuracy, there are several other metrics, they are: Precision, Recall, F1-score and ROC plot. “The accuracy is measured by the error rate, that is, the ratio of the number of misclassifications against the total number of classifications. The rates are calculated from the counts of data records correctly and incorrectly the classified by the classifier. The counts can be presented in a tabular form know as confusion matrix” (Du 2010, pp. 166). Accuracy is the value of (1 - error rate).

Based on different interpretation of confusion matrix, “Precision refers to the ratio of the number of true positive predictions to all positive predictions made by the classifier, while Recall is the ratio of the number of true positive predictions to the total number of actual true positive examples. High precision means few negative examples being predicted as positive. High recall means few positive examples are being predicted as negative” (Du 2010, pp. 203). F1-score is also called F-measure, which is $\frac{2 \times Recall \times Precision}{Recall + Precision}$ (Witten and Frank 2005, pp. 146). The F1 score reaches its best value at 1 and worst score at 0, it can be interpreted as a weighted average of the precision and recall.

The Receiver Operating Characteristic (ROC) curve plots “the number of positives included in the sample on the vertical axis, expressed as a percentage of the total number of positives, against the number of negatives included in the sample, expressed as a percentage of the total number of negatives, on the horizontal axis" (Witten and Frank 2005, pp. 142). Each curve in the plot represents the performance that the classifier deals with one class and closer a curve near the top left corner of the ROC plot, the better performance it is.

There is a relevant performance evaluation metrics for classifier evaluation named Area Under Curve (AUC), it is based on the ROC plot. It computes the area that under a curve in the ROC plot and the value of area indicates the performance (best value at 1) (Lobo, Jiménez-Valverde and Real 2008). The AUC is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance. Unlike Accuracy, it measures the classifiers in classifying a set of instance based on the degree to which they belong to the positive class, instead of actually assigning instance to classes.

The programming language used in this chapter is Python. All the data collection and selection, pre-process is written by thesis author except the code for SVM classifier estimation and evaluation. The basic code for SVM classifier estimation and evaluation is from scikit-learn Python package (Pedregosa et al. 2011).

The data pre-processing is described in section 4.4.4.1. Section 4.4.4.2 introduces the SVM classifier and utilizes Grid Search and Cross Validation methods to estimate the parameter set for the classifier from the pre-processed training dataset.

The evaluation result is described in section 4.4.4.3. Furthermore, the classification performance of dealing with “unfamiliar” subject data is evaluated in section 4.4.4.4.

4.4.4.1 Data Pre-processing

The data pre-process is a necessary sequence for machine learning, it consists of:

- Cleaning of Noisy Data.
- Standardization of Raw Data.
- Reducing Dimensions.

Noisy data is a major source of errors; it makes impact on the classification method which interferes with the classification process. In this experiment, the noise is mainly caused by the sitting habit of the subjects. Take the spine posture 3 and 4 raw data of subject 2 as an example, some zero value samples occur (zero values from all four channels), and the reason for this is because this participant prefers to keep the spine upright which causes very light contact force on the IntelliChair surface. The small amount of force is absorbed by the protective mat on the surface which prevents the FSR sensor underneath the mat from being triggered and generates zero value samples. All noisy data occurs in the spine posture, and those zero value samples are filtered out. The final sample numbers are 39162; consisting of 9075 spine posture samples and 30087 leg posture samples. Each class consists of around 2000 to 2550 samples. Unfortunately, the data of spine posture 3 and 4 from subject 2 is removed entirely because of the reason described above, but this does not affect the whole data for classification because both samples and labels are removed.

Data standardization aims to unify the samples in order to make the training process better behaved because of the improved numerical condition. Warren Sarle discussed the standardization options and their usage depends on the given dataset and the classification model (Sarle 2014). According to the discussion, the standardization strategy of this experiment is decided to be:

1. Chang mean value to zero
2. Rescale the data into the interval minimum -1and maximum 1 with midrange at zero

The reason for the strategy is because of the function based learning algorithm Support Vector Machine. It assumes that all features are centred around zero and the variances have same scales. If a feature had a variance that is significantly larger than others, this feature would dominate the decision function which interfering with the learning process. This is also the reason why the rescaling interval is [-1, 1] instead of [0, 1].

The reason why not rescales the data into mean 0 and standard deviation 1 is because of the feature of the data. This type of rescaling makes the learning algorithm behave well when the values of an individual feature follow Gaussian distribution. Normally, the random values of one channel from a subject follow the Gaussian distribution, but when the data from multiple subjects are merged, it becomes a set of Gaussian distribution that is mixed. The mean 0 and standard deviation 1 rescaling is not suitable for this type of data (Sarle 2014). It can be seemed from Figure 4-18 that there are three possible Gaussian distributions exist.

This is because of different body characteristics and sitting habits from multiple subjects, this is similar and typical situation for all the posture data.

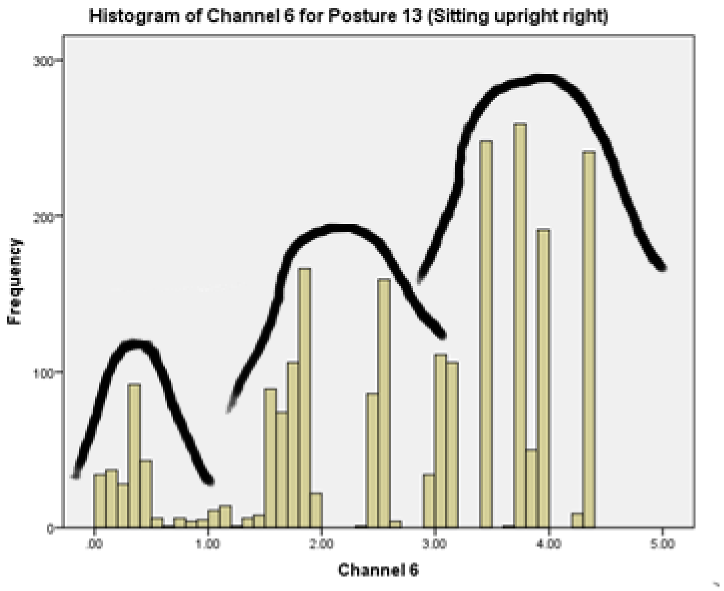
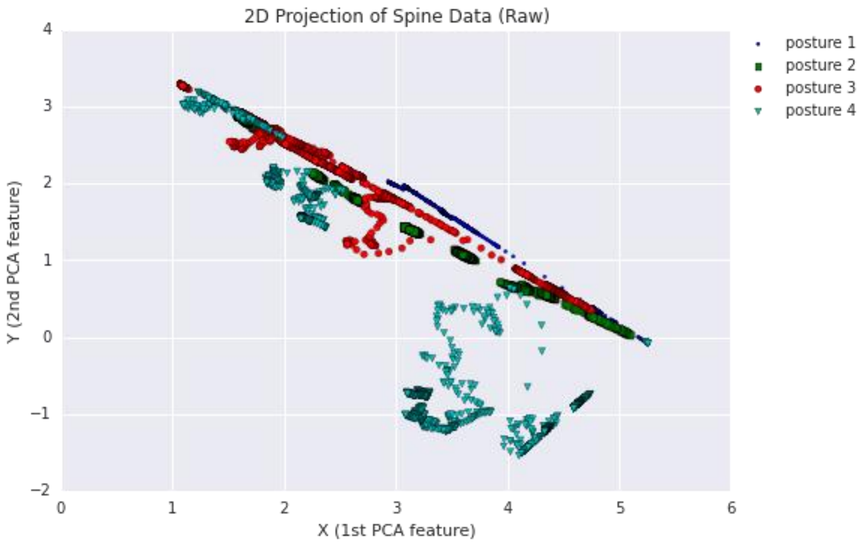


Figure 4-18 The Histogram of Channel 6 (detecting left hip) in sitting upright right posture. The black curves are three possible Gaussian distributions.

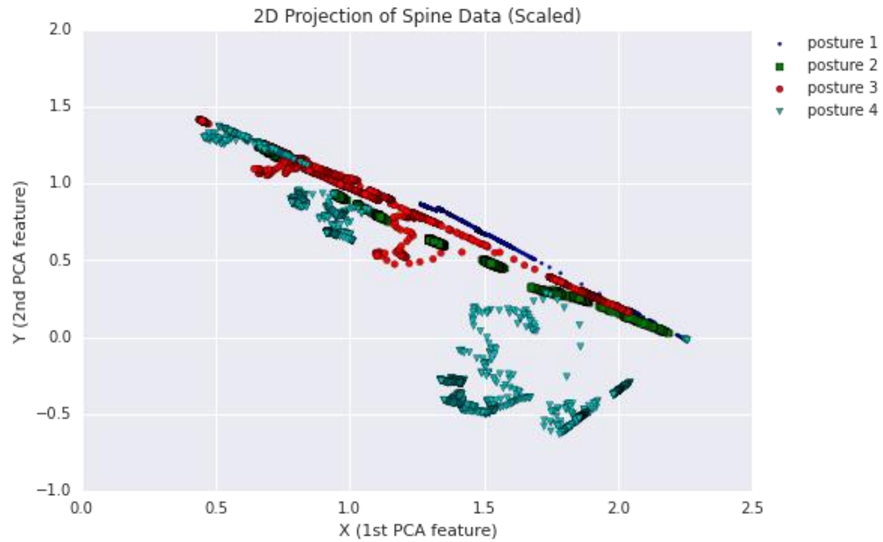
Dimension decrease is usually utilized in data pre-processing when there are large numbers of dimensions in the training dataset (i.e. face recognition, image processing) which might cause the curse of dimension problem. In this experiment, the sensors number only eight and their placements are in the critical position that was abstracted from previous research, and furthermore, the spine posture detection and leg detection are divided, so the dimension of each training dataset is four.

While there is no need to reduce dimensions, the Principle Component Analysis (PCA) method is still utilized, in order to achieve the data visualization that projects the four dimensional data into two-dimensional and three dimensional space. PCA aims to decompose a multivariate dataset into a set of components which is continuous and statistically independent. This set of components explains a maximum amount of the variance. In short, PCA is a transformation process that could project the high dimensional data into low dimensional. In this experiment, the explained variance ratios (sum equal 1) of first three components for the raw dataset are 0.45075424, 0.3231234 and 0.13959999 (total 0.91347763), which covers most variance of the raw dataset. The 2D and 3D dimensional posture data is shown in Figure 4-19.

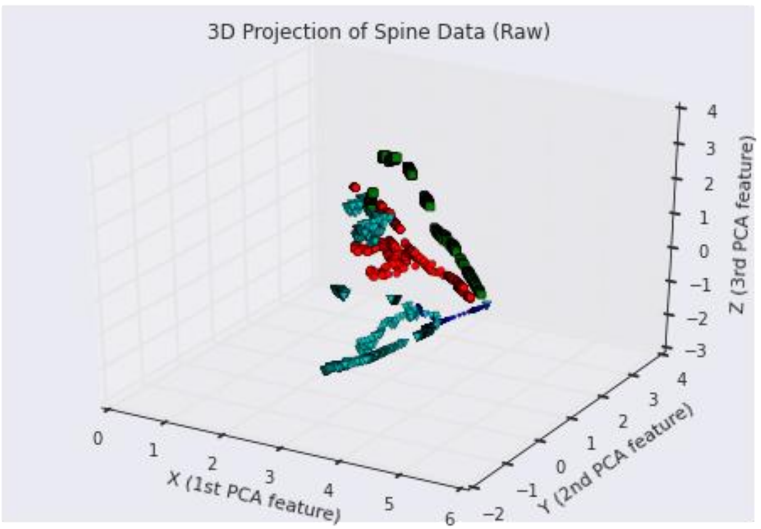
Figure 4-19 The 2D and 3D dimensional posture data (next 2 pages).
(a) The 2D projection of spine raw data.
(b) The 3D projection of spine raw data.
(c) The 2D projection of scaled spine data.
(d) The 3D projection of scaled spine data.
(e) The 2D projection of leg raw data.
(f) The 3D projection of leg raw data.
(g) The 2D projection of scaled leg data.
(h) The 3D projection of scaled leg data.


2D Projection of Spine Data (Raw)
posture 1
posture 2
posture 3
posture 4
Y (2nd PCA feature)
X (1st PCA feature)

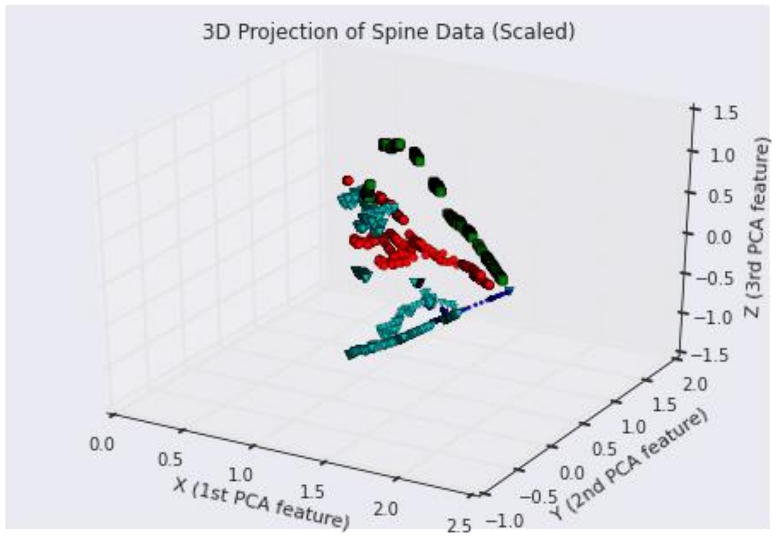
(a)


2D Projection of Spine Data (Scaled)
posture 1
posture 2
posture 3
posture 4
Y (2nd PCA feature)
X (1st PCA feature)

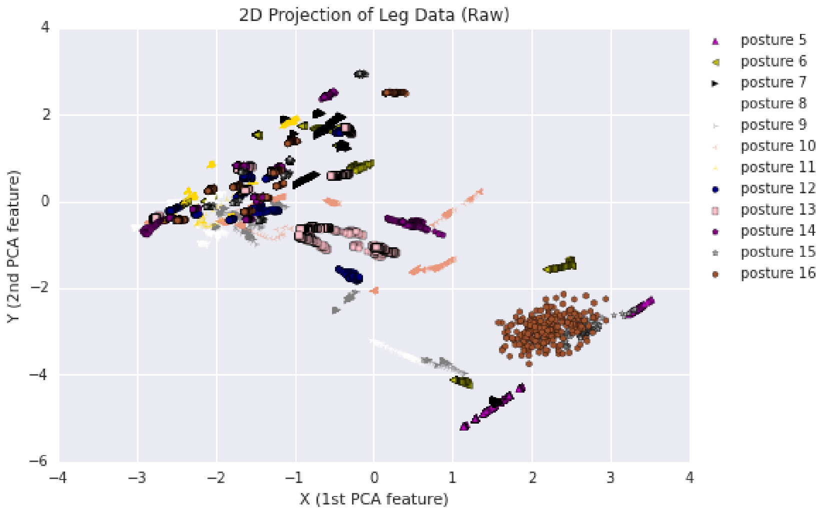
(c)


3D Projection of Spine Data (Raw)
X (1st PCA feature)
Y (2nd PCA feature)
Z (3rd PCA feature)

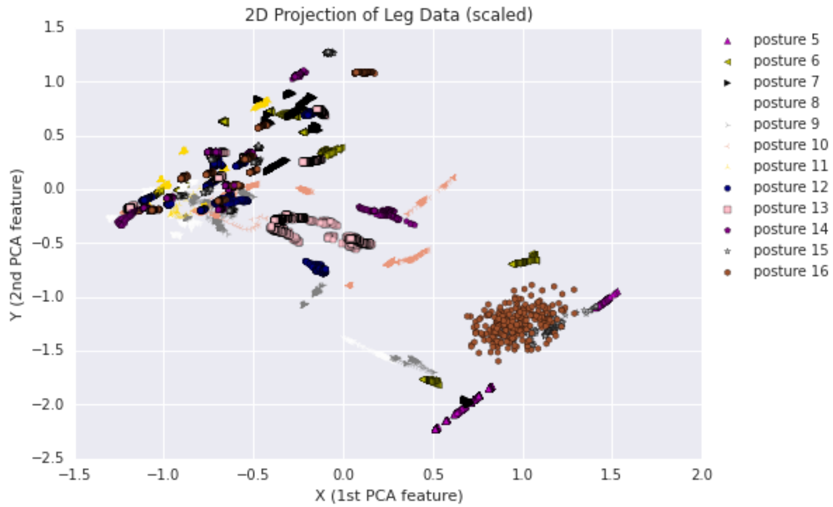
(b)


3D Projection of Spine Data (Scaled)
X (1st PCA feature)
Y (2nd PCA feature)
Z (3rd PCA feature)

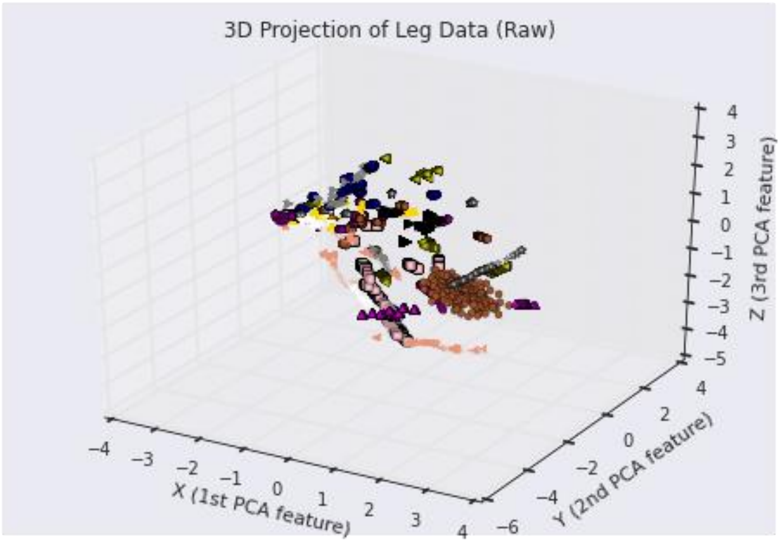
(d)


2D Projection of Leg Data (Raw)
Y (2nd PCA feature)
X (1st PCA feature)
posture 5
posture 6
posture 7
posture 8
posture 9
posture 10
posture 11
posture 12
posture 13
posture 14
posture 15
posture 16

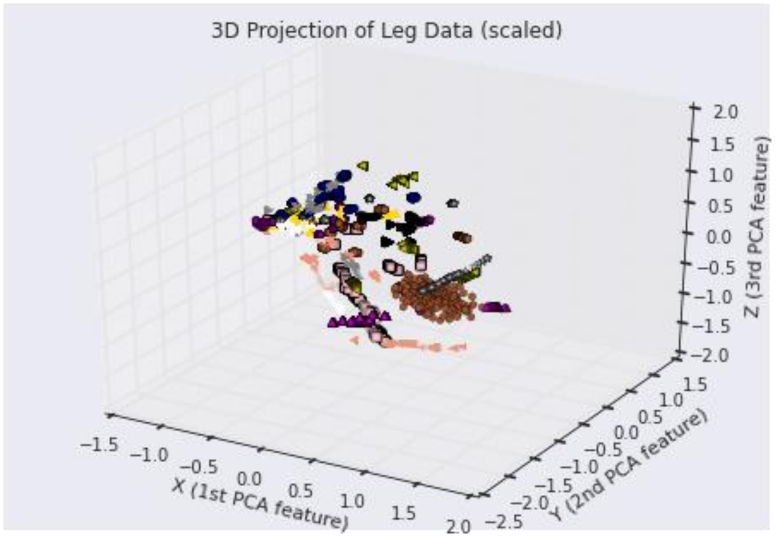
(e)


2D Projection of Leg Data (scaled)
Y (2nd PCA feature)
X (1st PCA feature)
posture 5
posture 6
posture 7
posture 8
posture 9
posture 10
posture 11
posture 12
posture 13
posture 14
posture 15
posture 16

(g)


3D Projection of Leg Data (Raw)
X (1st PCA feature)
Y (2nd PCA feature)
Z (3rd PCA feature)

(f)


3D Projection of Leg Data (scaled)
X (1st PCA feature)
Y (2nd PCA feature)
Z (3rd PCA feature)

(h)

Figure 4-19 indicate the feature of the data:

- The scaling process does not change the sample distribution in the space.
- The samples of different classes in spine dataset are sparsely distributed in the space which means the classification on spine classes could have very good performance.
- The samples of different classes in leg dataset are crowded together rather than sparse in the space.

Those data features convey the spatial distribution of the posture data based on mixture of different classes. This helps the determination of posture classification strategy. According to the data feature, the Support Vector Machine classification algorithm is selected for the sitting posture classification. The reasons include (Gunn 1998):

1. Effective in the situation that number of data dimensions is lower than the number of samples. In this case, there are 4 dimensions in the data, which is much lower that the sample numbers.
2. Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient. Considering the sitting posture classification component is deployed on the micro-PC, memory efficient is essential in this part.
3. Versatile: different Kernel functions can be specified for the decision function for different classification strategy. It can be seem from the data features that the classification is required to recognize the samples when the samples are crowded together. In this situation, SVM is able to accomplish the task with

flexible choice of kernel function. The detail about kernel function selection is discussed in the next section.

Reason 1 and 2 is based on the numeric sensitive feature of the SVM, and reason 3 is extends the usage of the SVM from linear classification to non-linear classification.

Those features of SVM are not only the feature of SVM but also its advantages compare with other classification methods. Other classification methods like Decision Trees, k-Nearest Neighbours and Artificial Neural Network are also famous, but they are based on different classification approach and they are less suitable for the sitting posture classification in this research compare with SVM.

For Decision Tree and k-Nearest Neighbours, they are instance or sample based classification. Their classification result is high dependent on the training dataset. The larger the dataset is, the higher the accuracy will be. But this also causes an extreme large trained model and it is possible cause a serious overfitting problem.

The Artificial Neural Network is a probability theory based classification algorithm. According to Mota and Picard's research (Mota and Picard 2003); it could achieve higher classification accuracy on new upcoming subject data (not included in the training data). But considering the sitting posture classification component is running on the micro-PC (Raspberry Pi) with limited computation power (limited processor power and limited memory), the SVM is better because SVM is memory efficient which is the reason 2 in the discussion of SVM.

The detail of the SVM's theory and features is discussed in next section. Furthermore, the details on how to implement the SVM classification on the collected sitting posture data is also discussed include the selection of the SVM kernel function, the parameter estimation and optimization of SVM classifier.

4.4.4.2 Support Vector Machine Classifier Estimation

The Support Vector Machine (SVM) is used as a linear classifier for binary classification problem which outputs a Yes or No result that determines whether a given sample belongs to one class or not (Cortes and Vapnik 1995). The mathematics of SVM is described as:

- A known training dataset with instances and class labels paired:
  $(x_i, y_i), i = 1, ..., l, x_i \in R^n, y \in \{-1\ , 1\}^l$
- The solution for the class separation by maximising the distance between instances in the class and hyperplane:

$$
\begin{array}{ll}
\min\limits_{\mathbf{w},b,\xi} & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}\xi_i \\
\text{subject to} & y_i(\mathbf{w}^T\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \\
& \xi_i \geq 0.
\end{array}
$$

(Hsu, Chang and Lin 2010)

The trick to solve the problem is to map the vector (instance) x_i into higher dimensional space, and this is done through kernel function $\emptyset$. The whole process is called Kernel Trick (see section 4.4.3). After the mapping, classes which are not able to be linearly classified in lower dimension are able to be linearly classified in the higher dimension. And the function that could accomplish this mapping is called a kernel function. The penalty parameter $C > 0$ is for the error term, which is a universal parameter for all possible kernel functions.

There are four basic kernel functions: linear, polynomial, sigmoid and radial basis function (RBF). The RBF kernel is most commonly used because its parameter is less than sigmoid and polynomial (Hsu and Lin 2002) which make it easier to be estimate. More importantly, it can handle the case when the relationship between class labels and features of instance is nonlinear. Unlike linear kernel, it has only one parameter γ and the kernel is represented as

$$K(x_i, x_j) = exp(-\gamma \left\| x_i - x_j \right\|^2), \gamma > 0$$

Now, the decision which class a instance belongs to relies on the decision function

$$f(\mathbf{x}) = \sum_{i=1}^{N_s} \alpha_i y_i \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\right) + b$$

In this function, N_s is the number of support vectors, (x_i, α_i, y_i) is the i -th support vector. The distance between the instance and the hyperplane is computed and it determines the class that the instance belongs to.

This is the process for binary classification, but in the real world, nevertheless, the SVM could be used for multi-class classification. It use one-vs-rest strategy, and creates a set of SVM classifiers. Each classifier deals with one class. But the results can be combined to achieve multi-class classification.

The classification performance relies on the selection of Kernel Function of SVM and the choice of the selection is based on the training data. If the classes in the given training data are able to be classified linearly, then the linear Kernel Function is able to fulfil the task. But if the data in the classes are surrounded or mixed with other classes, the RBF Kernel Function is preferred for the non-linear classification task.

According to the initial work of the thesis author (Fu and Macleod 2014), the RBF is more suitable in this experiment compared with linear kernel, so the hyperparameter $(C > 0, \gamma > 0)$ is required to be determined.

The γ selection dramatically affects the performance of the classifier because it defines how far the influence of a single training instance could reach; the lower the value is, the further the distance. Conversely, the higher the value is, the closer the distance. It is the inverse of the radius of influence of instance selected by the model as support vectors.

The $C > 0$ penalty parameter determines the simplicity of the model decision surface which tackles the misclassification of training instances. The lower the *C* value is, the smoother the surface becomes which also means could include outlier instances from other classes. Meanwhile, with a higher *C* value, the surface tends to classify all training instances correctly by giving the model freedom to select more samples as support vectors. Figure 4-20 shows the process detail includes both training and classification stage, and the roles of the kernel function and its parameters in the SVM classification process.

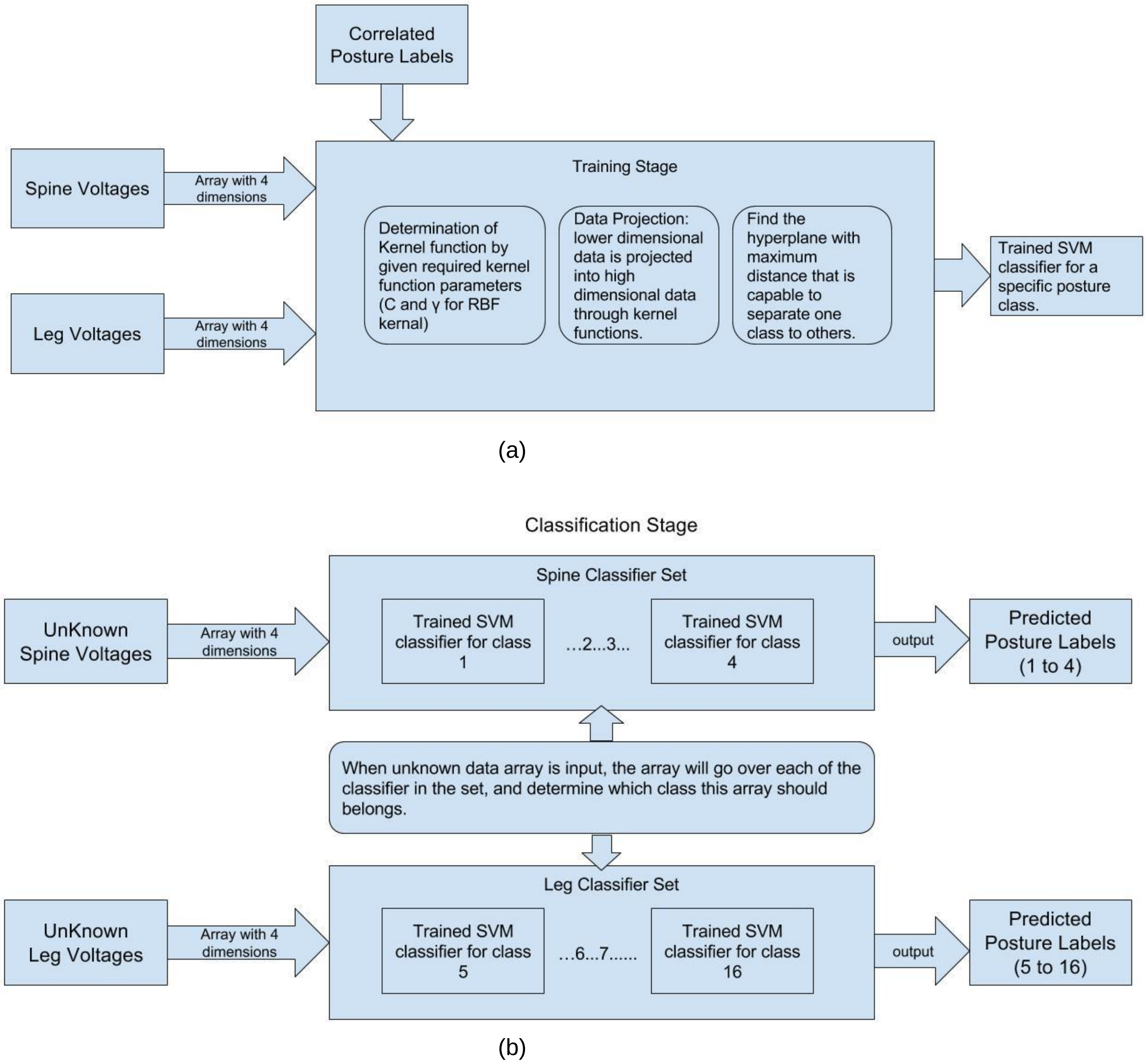


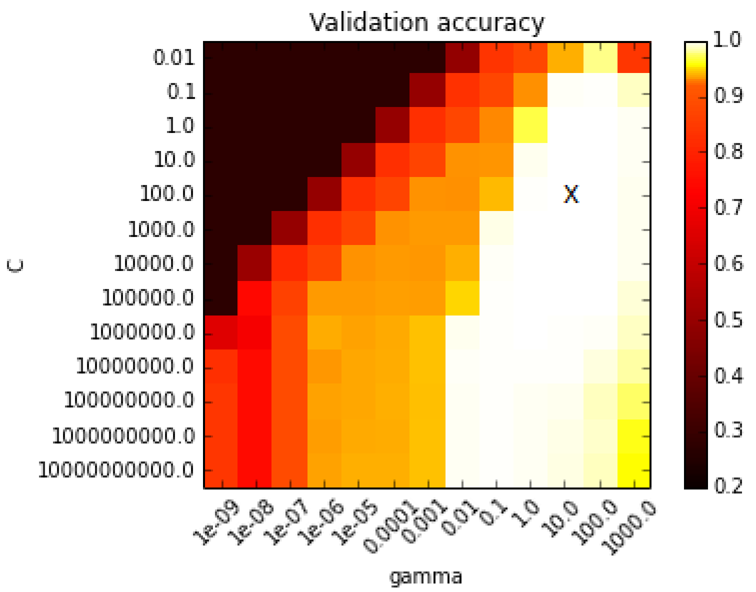
Figure 4-20 The demonstration of the SVM training and classification stage.
(a) is the training stage, data arrays and correlated posture labels is input and through three steps (kernel function setup, data projection and support vectors calculation for best hyperplane), one SVM classifier of one specific class is trained. Total 16 classifiers will be divided into spine classifier set and leg classifier set.
(b) is the classification stage which is when unknown data arrays is input, it will go through each of the classifier in the set and its class will be determined.

It can be seem from the figure that the selection of kernel function and its parameter is important in SVM classification. But in the practical situation, the span of the $C > 0$ and $\gamma > 0$ is very large. In the Hsu's paper (Hsu, C, 2010), the possible range covers $C = 2^{-5}, 2^{-3}, ..., 2^{15}, \gamma = 2^{-15}, 2^{-13}, ..., 2^{3}$. In this experiment, the purpose is covering a large interval of hyper-parameters(C, γ). The range is proposed to be$(C = 10^{-2}, 10^{-1}, ..., 10^{10}, \gamma = 10^{-9}, 10^{-8}, ...10^{3})$. The reason for the change from logarithm base from 2 to 10 is because the convergence of the hyperparameter set for the posture classification might be distributed in a much large area compared with the common initial guess. The logarithm base of 10 is selected in order to covers a larger range with less number of steps compared with logarithm base of 2.

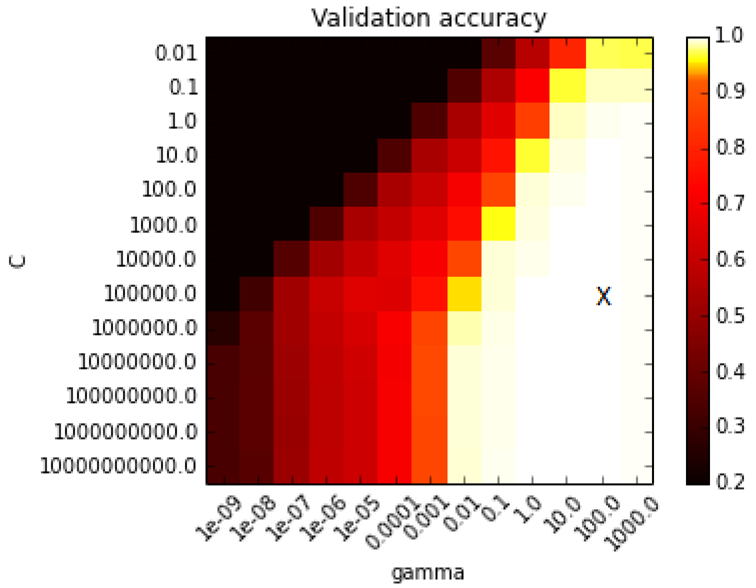
According to the description in the beginning of section 4.4.4, every combination within the hyperparameter (C, γ) is used to train a SVM classifier in the Grid Search process stage, in order to identify the parameter set that gave the classifier the best performance. The accuracy is used as metrics for the model selection. Through the experiment, the initial results of the Grid Search process are:

- Spine SVM parameters: $C = 10^{3}, \gamma = 10$
- Leg SVM parameters: $C = 10^{5}, \gamma = 100$

The parameter set above achieves 99.8% for spine posture classification and 99.8% as well for leg posture classification. Figure 4-21 indicates the accuracy distribution in the hyperparameter set space.



(a)



(b)

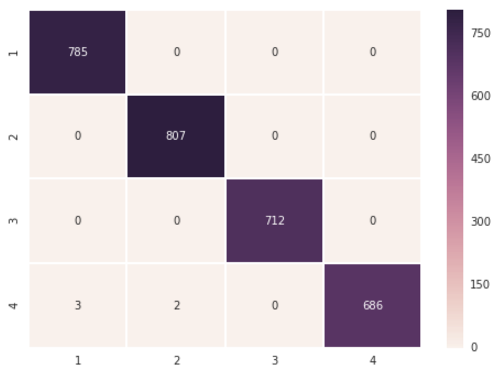
Figure 4-21 The x axis is the gamma parameter values, y axis is the C parameter values. Different color indicates different accuracy that based on the particular parameter set. The lighter the color is, the higher the accuracy.
X marker in the figure is the location of the parameter set with the highest accuracy.
(a) The spine classifier parameter set.
(b) The leg classifier parameter set.

Since both classification accuracies reach 99.8%, the result seems promising in the training stage. The next step is to evaluate the model by using a dataset which is unknown to the model.

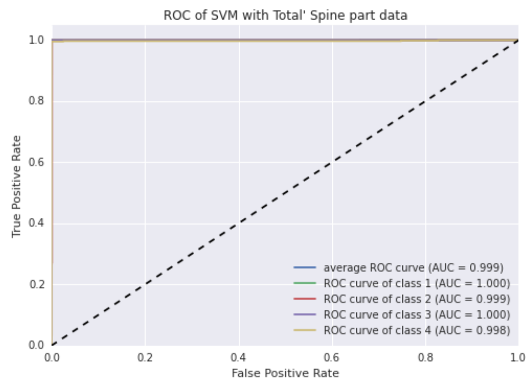
The evaluation result for the spine SVM classifier is listed in Table 4-10, and the confusion matrix and ROC plot for spine classifier's evaluation is shown in Figure 4-22. The evaluation result for the leg SVM classifier is listed in Table 4-11, and the confusion matrix and ROC plot for leg classifier's evaluation is shown in Figure 4-23.

	Precision	**Recall**	**F1-score**	**Support**
Posture 1	1.00	1.00	1.00	785
Posture 2	1.00	1.00	1.00	807
Posture 3	1.00	1.00	1.00	712
Posture 4	1.00	0.99	1.00	691
Avg / Total	1.00	1.00	1.00	2995

Table 4-10 The classification report of the spine SVM classifier to 2 decimal places.



(a)

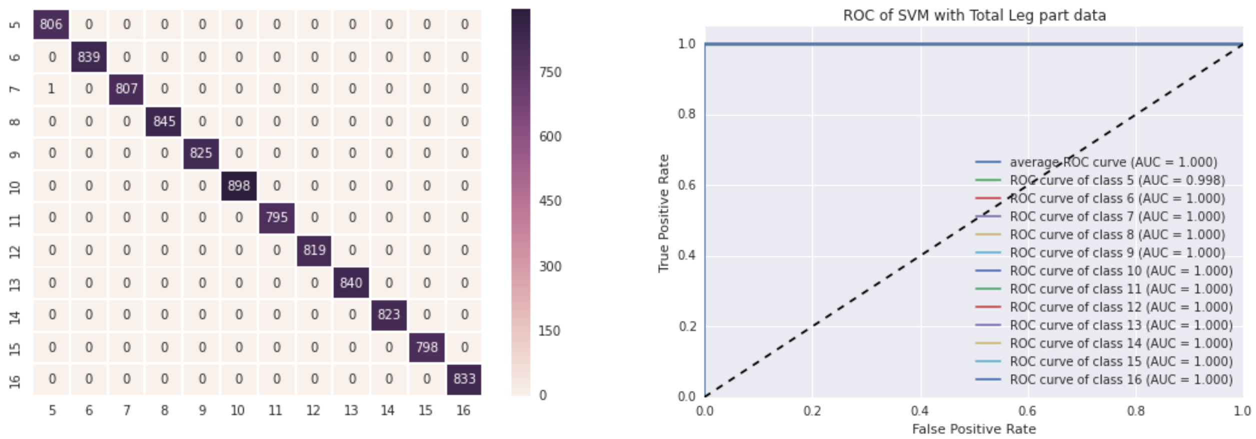


(b)

Figure 4-22 The confusion matrix (a) and ROC plot (b) for the spine SVM classifier. In (a), the x axis is the predicted label; y axis is the true label.

	Precision	Recall	F1-score	Support
Posture 5	1.00	1.00	1.00	806
Posture 6	1.00	1.00	1.00	839
Posture 7	1.00	1.00	1.00	808
Posture 8	1.00	1.00	1.00	845
Posture 9	1.00	1.00	1.00	805
Posture 10	1.00	1.00	1.00	898
Posture 11	1.00	1.00	1.00	795
Posture 12	1.00	1.00	1.00	819
Posture 13	1.00	1.00	1.00	840
Posture 14	1.00	1.00	1.00	823
Posture 15	1.00	1.00	1.00	798
Posture 16	1.00	1.00	1.00	833
Avg / Total	1.00	1.00	1.00	9929

Table 4-11 The classification report of leg SVM classifier to 2 decimal places.



(a) (b)

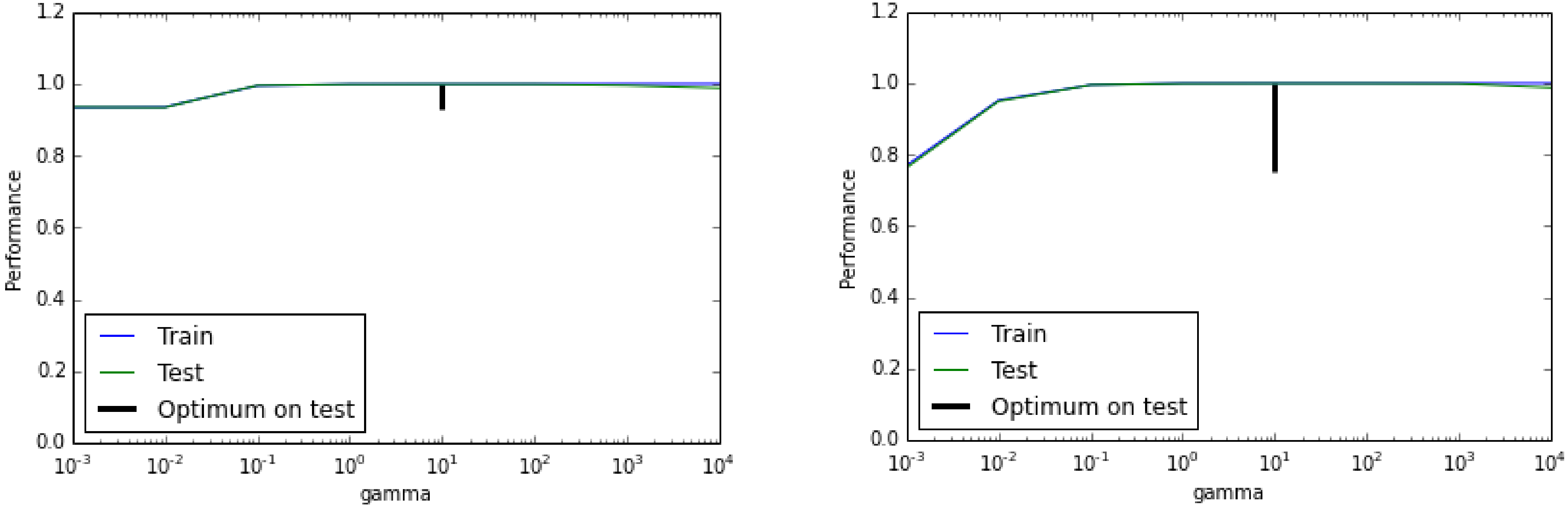
Figure 4-23 The confusion matrix (a) and ROC plot (b) for the leg SVM classifier. In (a), the x axis is the predicted label, y axis is the true label.

The overall classification accuracy of spine classifier is 0.998, while the overall classification accuracy of leg classifier is 0.999899284923. Because of the decimal place numbers, some the result in Tables and Figures above are shown as 1.00 or 1.000.

4.4.4.3 Evaluation of the Classifier

The classification result in the evaluation stage seems to be very good, but it raises another question, is the model over-fitted? In order to answer the question, the validation curve is drawn for both spine and leg classifier and Accuracy is used as metrics.

Because the γ value has higher weight in the hyper parameter set compare with C (because γ represents the kernel function which is the core part for mapping), and considering γ has smaller range compared with C according to Figure 4-21, the next step is to use the grid search method again, but focus on γ value, and the gird search space for γ value is $\gamma = 10^{-3}, ..., 10^{4}$ (according to Figure 4-21 b). The validation curve for over-fitting test is shown in Figure 4-24.



(a) (b)

Figure 4-24 The validation curve of two SVM classifiers.
(a) The accuracy validation curve of SVM classifier for spine
(b) The accuracy validation curve of SVM classifier for leg.
The y axis is the Accuracy.

According to Figure 4-24, both training values and test values converge at the point where γ = 10. The test curve and train curve diverge for the spine classifier at 100, while the two curves diverge at 1000 for the leg classifier. So the interval for the γ value is:

- For spine classifier: $\gamma \in [10, 100]$
- For leg classifier: $\gamma \in [10, 1000]$

The interval means that γ values in this range should have the best performance for the sitting posture classifiers. In order to simplify the setup, the γ values are set to be 10 in both classifiers. As described in section 4.4.4.2, γ defines how far the influence of a single training instance could reach; the lower the value is, the further the distance. In order to maximise the decision boundary, the lower γ is preferred.

Combined with the C discussed previously (determines the smooth of the decision boundary), the final hyper parameter set for the SVM classification method is:

- For spine classifier: kernel function is RBF, and $(\gamma = 10, C = 10^3)$.
- For leg classifier: kernel function is RBF, and $(\gamma = 10, C = 10^5)$.

One thing need to be clarified is that in the optimisation process in this section, the dataset that is used for evaluation is from the same subject dataset, not completely new dataset from unknown subjects. Reader can consider the evaluation dataset as familiar or similar dataset with training dataset, but not the same dataset that is used for training. The difference between those two can be referred to the content in the beginning of section 4.4.4.

4.4.4.4 Further Evaluation on Support Vector Machine Classifier

The evaluation result for the SVM classifier in the previous section is good in both validating and testing stage. This result is proofed by both scikit-learn Python machine learning package and Java based Data Mining software WEKA. But is it a reliable performance when dealing with completely "unfamiliar" data? Is the SVM classifier parameter set generated by the whole data suitable when building SVM classifiers (spine and leg) for a single subject?

In order to further investigate the questions, the next step is to utilize the Grid Search method on the individual subject. The train, validate and test procedure is the same, but the data is from only one subject. The reason for this process is to explore the (C, γ) parameter set difference between individual subject data and the whole data.

Grid Search for parameter set for different Subject Spine Data										
	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Subject 6	Subject 7	Subject 8	Subject 9	Subject 10
C	0.01	1.0	0.01	0.01	0.01	0.01	10	0.1	0.01	0.01
γ	1	1	1	1	100	1	1	100	1	10

Grid Search for parameter set for different Subject Leg Data										
	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Subject 6	Subject 7	Subject 8	Subject 9	Subject 10
C	0.1	0.01	0.01	0.01	0.1	0.1	0.01	0.1	0.01	0.01
γ	100	100	10	100	10	100	10	10	100	10

Table 4-12 The parameter set for different subject data.

According to Table 4-12, the *C* values for individual data are clearly lower compare with the *C* value from the whole data while γ is similar. From the definition of the SVM (see section 4.4.4.2), a lower *C* means a smoother decision surface, that makes it easier to misclassify an outlier.

In order to find a better value for *C*, the parameter set will be tested again, considering both individual subject parameter and the whole subject parameter. The difference for this test is that instead of using the whole dataset, one of the subjects will be left as a test dataset, and the remaining nine subjects' dataset will be used for training and validating. This is also an evaluation to see how the classifier works when dealing with an "unfamiliar" subject data. Considering the interval of the parameter set, the interval for the parameter set is:

- Spine classifier: *C* ranges from 0.01 to 1000, γ ranges from 1 to 100.
- Leg classifier: *C* ranges from 0.001 to 100000 γ ranges from 10 to 100.

Spine Classifier Parameter Set							
C							
γ		0.01	0.1	1	10	100	1000
	1	0.786	**0.807**	0.755	0.757	0.753	0.683
	10	0.766	0.802	0.779	0.773	0.772	0.765
	100	0.530	0.617	0.649	0.649	0.650	0.652

Leg Classifier Parameter Set									
C									
γ		0.01	0.1	1	10	100	1000	10000	100000
	10	0.415	**0.423**	0.418	0.414	0.421	0.417	0.417	0.417
	100	0.308	0.196	0.202	0.203	0.203	0.203	0.203	0.203

Table 4-13 Classifier performance (accuracy) in different parameter set, tested by each individual subject (Maximum accuracy is highlighted in bold).

As listed in table 4-13, the accuracy when the classifier is dealing with an “unfamiliar” subject’s data is not satisfactory, especially the leg classifier. Although the result is poor, it can be concluded from tables that when the *C* and γ value are low, the performances are relatively better. For γ value, when it increases, the classifier performance usually decreases. Especially from 10 to 100, the performance drops dramatically. So currently, the parameter sets that have the best accuracy to deal with “unfamiliar” subjects are are:

- Spine classifier parameter set: $(C = 0.1, \gamma = 1)$
- Leg classifier parameter set: $(C = 0.1, \gamma = 10)$

Because the classifier is dealing with new subject data and Table 4-13 only shows the average accuracy of each subjects. It does not reflect the real performance that it deals with each of the postures of each subject. Another test is therefore performed with the SVM classifier with the optimised parameter set and applied to each individual subject data as test data, using the Area Under Curve (AUC) value as the indicator.

As discussed in the beginning of section 4.4.4, the accuracy and AUC measure are different qualities of the classifier. Accuracy ignores probability estimations of classification in favour of class labels (based on fixed threshold) while ROC curves show the trade-off between false positive and true positive rates (the curve is generated through a group of threshold). In that situation, AUC is a better indicator of classification performance compare to the fixed threshold which hided the threshold information (Lobo, Jiménez-Valverde and Real 2008). Thus, the AUC measure and the Accuracy are not necessary correlated (High AUC does not mean high accuracy, it is the same conversely).

The AUC value of Spine classifier with C = 0.1 gamma = 1													
Posture		Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Subject 6	Subject 7	Subject 8	Subject 9	Subject 10	Avg	STD
	1	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1	0
	2	1.0	1.0	1.0	1.0	1.0	1.0	0.993	1.0	1.0	1.0	0.999	0.002
	3	**0.333**		1.0	1.0	0.971	1.0	**0.361**	0.998	1.0	0.988	0.850	0.285
	4	0.959		0.708	1.0	1.0	1.0	0.550	0.951	1.0	1.0	0.907	0.164
	Avg	0.742	1.0	0.999	0.962	0.998	0.958	0.783	0.991	1.0	0.969	0.940	0.096

Table 4-14 The AUC measure of the posture and the subject as test data for spine classifier.
The blank cell is because the filtered noisy values.
The values that are less than 0.5 are highlighted in bold.

The AUC value of Leg classifier with C = 0.1 gamma = 10													
Posture		subject 1	subject 2	subject 3	subject 4	subject 5	subject 6	subject 7	subject 8	subject 9	subject 10	Avg	STD
Sitting on Edge	5	0.947	1.0	1.0	1.0	1.0	1.0	1.0	0.818	1.0	1.0	0.977	0.058
Crossing Right Leg on Left Leg	6	0.910	1.0	1.0	1.0	0.910	1.0	**0.354**	1.0	**0.163**	**0.01**	0.735	0.395
Crossing Left Leg on Right Leg	7	0.590	1.0	1.0	1.0	1.0	1.0	**0.405**	1.0	1.0	1.0	0.900	0.216
Sitting Forward	8	**0.089**	**0.198**	**0.006**	**0.182**	1.0	**0**	**0.001**	0.632	**0.091**	0.909	**0.311**	0.387
Sitting Forward Left	9	0.720	1.0	1.0	1.0	1.0	1.0	0.670	1.0	**0.361**	0.702	0.845	0.222
Sitting Forward Right	10	**0.111**	0.828	1.0	0.818	**0.205**	1.0	0.997	1.0	1.0	1.0	0.796	0.344
Sitting Upright	11	0.579	0.910	**0.095**	0.906	0.908	**0.438**	**0.0**	0.915	1.0	**0.26**	0.601	0.380
Sitting Upright Left	12	**0.357**	0.568	**0.455**	1.0	1.0	**0.181**	0.986	1.0	0.818	0.909	0.727	0.310
Sitting Upright Right	13	**0.201**	0.988	0.844	0.991	0.683	**0.499**	0.996	**0.312**	1.0	0.904	0.742	0.304
Leaning Back	14	**0.463**	0.096	0.536	**0.356**	0.896	1.0	0.771	**0.252**	0.908	1.0	0.628	0.330
Leaning Back Left	15	0.756	**0.276**	0.909	1.0	1.0	0.960	**0.461**	**0.0**	1.0	0.909	0.727	0.357
Leaning Back Right	16	1.0	0.596	0.729	0.524	**0.195**	1.0	0.636	**0.286**	0.88	**0.036**	0.588	0.334
	Avg	**0.553**	0.667	**0.631**	0.797	0.832	0.729	**0.601**	0.688	0.809	0.701	0.701	0.092

Table 4-15 The AUC measure of the posture and the subject as test data for leg classifier.
The values that are less than 0.5 are highlighted in bold. Some values that is higher than 0.5 in bold is for the discussion below.

Tables 4-14 and 4-15 show the AUC measures when SVM is using optimized parameter set to test across each individual subject’s data. The higher value means better classification result for these classes. In table 4-14 for spine data, it is clear that all spine postures are classified very robustly, except the posture 3 (body leaning back£ for subjects 1 and 7.

In contrast, Table 4-15 for leg data shows that the classification results for the leg postures have large deviations for AUC measures. The bold ones are the AUC measures lower than 0.5 (not including the ones in Subject Avg row) which indicate AUC measures that is not satisfactory. Table 4-15 provides more detailed information about AUC measures. The readers is able to trace the AUC measures not only the overall AUC measure for subjects or leg postures, but also the AUC measures of each leg posture of each subject.

Through each subject based columns, it can be seen that there are one or several low value AUC measures for each subject, which means the correlated leg postures are hard to be recognized by system, and the reason is probably because the relatively similar pressure distribution data with other leg posture data from one subject, and this causes the SVM misclassified.

According to Table 4-15, the most robust classification are the posture 5 (sitting on edge) and posture 7 (crossing left leg on right leg). The worst classification is the posture 8 (sitting forward), but within the row of result, there are still two AUC measures that are very high (0.907 from subject 10 and 1.0 from subject 5).

The high variation of AUC measures can be explained from individual differences between both subjects and the sitting habits. From subject's point of view, the worst average AUC measures come from subject 1, subject 3 and subject 7. Subject 1 has the highest weight and height while subject 3 has the second highest height; the weight difference between them is large (13 KG difference). Subject 7 has the equal lowest height and slim body (low weight), which make some of his posture is not well classified. The body characteristics differences cause this classification results.

Inheriting the discussion above, the subject 2 is same height and 10 KG lighter, but the AUC measure is different, and this can be explained from the sitting habit point of view. Take the lowest three AUC measures of subject 6 as example, which are posture 11 (sitting upright), posture 12 (sitting upright left) and posture 13 (sitting upright right). Those three classes have low AUC measures but subject 6 has good AUC measures (1.0, 0.96, 1.0) for posture 14 (leaning back), posture 15 (leaning back left) and posture 16 (leaning back right). It probably means subject's sitting upright posture is actually more close to leaning back posture compare with other subjects, and the AUC measure of subject 6's posture 8 (sitting forward) is 0 while subject 6's posture 5 (sitting on edge) AUC measure is 1.0, and this is probably because subject 6's preferred sitting habit for posture 8 (sitting forward) is actually performed like posture 5 (sitting on the edge). This is the same situation for subject 7 (AUC measure is 0.001 for sitting forward, AUC measure is 1.0 for sitting on edge).

The situation is different for subject 8, subject 8’s worst result is for leaning back postures (posture 14 to 16) and the best result is posture 11 (sitting upright) and posture 12 (sitting upright left), and this probably means subject 8’s leaning back postures is more close to the sitting upright postures.

4.4.5 Summary of Sitting Posture Classification Experiment

Section 4.4 described the detail of sitting posture classification experiment including the statistical analysis and pre-process on posture data, the estimation of SVM classifier and classification performance evaluation when dealing with data from both “familiar” and “unfamiliar” subjects.

The classification performance shows that the SVM based classifier is robust to “familiar” subject data (accuracy is 99.8% with spine postures and 99.9% with leg posture) which is shown in the end of section 4.4.4.2. When dealing with “unfamiliar” subject data, the accuracy are 80.7% for spine posture classification and 42.3% for leg posture classification (see Table 4-13 in section 4.4.4.4).

The combination of both sitting habit and body characteristics differences causes the high variation of classification result between different postures; furthermore, it affects the overall classification accuracy when the classifier is dealing with “unfamiliar” subjects.

Another factor that emphasizes the poor classification result for “unfamiliar” subjects is the hardware setup. Because the FSR sensors are deployed on the critical position of the chair surface, so that the pressure data is different when individual sitting habit is different this makes impact on the “unfamiliar” subject data classification. But its advantage is that once the classifier is trained by one subject, it makes excellent fit and classification result for that subject. Compared with the classification results of Tan’s research (according to Table 2-1, Sensing chair achieves 79% with “unfamiliar” subject, 96% with “familiar” subject with 10 postures.) (Slivovsky and Tan 2000, Tan, Slivovsky and Pentland 2001, Tan 1999), their system is more capable to deal with “unfamiliar” subject data is much better because of the utilization of pressure sensor array mat. The input data for their system is actually a pressure image, which is less sensitive to the individual difference because of the high resolution of data.

4.5 Summary

Chapter 4 includes the details of the four experiments that answer the first four research questions discussed in section 3.2. Through those experiments, the IntelliChair system is built gradually.

The first experiment determines the pressure sensor for IntelliChair system between Force Sensing Resistor (FSR) and conductive foam. The FSR is selected because of the lower measurement deviation and FSR sensor is able to maintain its measurement stability under the long time and high pressure condition according to the result in section 4.1.2.

The second experiment built a data visualization system to convey both pressure data and skeleton data in order to evaluate their usability. Kinect sensor is discarded due to its stability in skeleton detection for sitting posture because of its depth information based sensing nature. This nature causes the joint drifting problem which occurs in the lower part of the body (hip, thigh, leg, and knee). This is not acceptable for system that aims to detect sitting posture because in sitting situation, several joints (knees and hips) appear overlapped in the depth information (see section 4.2.2). According to the experiment result, the pressure sensing component is more reliable because it can accurately and quickly response to the user’s sitting posture change in the pressure information visualization system.

In the third experiment, the frequency characteristic of micro motion changing in sitting posture is analysed based on the pressure sensor system of IntelliChair. Despite there being some changes on the hardware platform (from Arduino to Raspberry Pi), the Python code based program is transferable due to the platform independent feature of Python, so there are no effect on the experiment result, and Raspberry Pi is the main platform for the rest of the thesis. The result of this experiment determines the sampling frequency of the IntelliChair system for the activity experiment in chapter 5. According to the result in section 4.3.3, the 6.2 Hz sampling frequency of IntelliChair system enable the system to detect posture changes in the situation that the subject is performing “normal” activity. The analysis is achieved by the implementation of the Fourier Transform that reveals the frequency feature from the collected time series based data.

Based on the built IntelliChair pressure sensing system, the experiment for sitting posture classification is performed. This experiment estimates two Support Vector Machine classifiers, one deal with spine posture classification and the other one deal with leg posture classification. Both classifiers are trained and evaluated by the collected sitting posture data with pre-defined posture labels. The evaluation result for the classifier is that the classifiers are sensitive to “unfamiliar” subject data but robust to the “familiar” subject data (see section 4.4.5). It indicates that IntelliChair requires a training stage before deployment of sitting posture detection and it is capable for the situation of individual usage such as sitting posture monitoring in office, etc.

The importance about the results above is that those results make sure the IntelliChair fulfil the requirements which are: non-intrusive sensing, low cost on sensor and the ability to recognize sitting posture. All those result helps in establishing a pressure sensing system for the aim of this thesis: sitting activity recognition. The content of the next chapter (chapter 5) is about the activity analysis and modelling. The aim of chapter 5 is to recognize the activity through the posture sequence which is the time-ordered output of the trained sitting posture classifiers in section 4.4.

Chapter 5 Result of Sitting Activity Modelling and Recognition

In section 4.4, a sitting posture classifier was developed and its task is to classify the incoming pressure data from 8 FSR sensors into 16 postures, four spine postures and twelve leg postures. This classification process not only allows the IntelliChair users to know their sitting postures, but also produces a sequence of sitting postures over time. This is now used as an input to estimate the user's activity state.

This chapter explains the sitting activity recognition component which is based on the input of the sitting posture classification output from section 4.4. Figure 5-1 shows the architecture of the IntelliChair system in deployment stage.

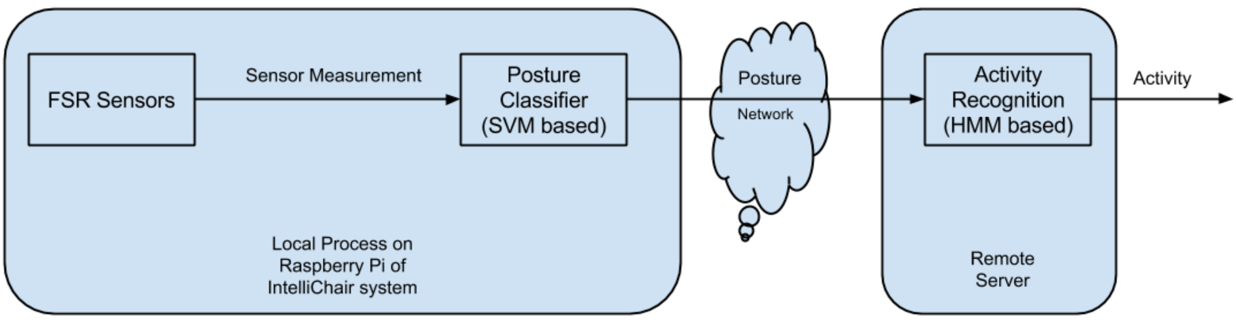


Figure 5-1 The system architecture of IntelliChair in deployment stage.

It is relevant to highlight that a separate sitting posture classification layer was needed in this thesis. There are three reasons for the separation:

- The output of the posture classification process provides simplified sitting posture information for the Activity modelling and recognition process which is described in this chapter. Thus, the activity state could be represented by sequence of posture.
- The posture classification layer can be store and perform the task locally on the Raspberry Pi micro PC where the sensors are directly connected. This local process could offer the IntelliChair user an instant feedback about their sitting posture.
- The simplified posture information reduces the data transmission traffic because the posture modelling is carried out locally with only classified pressure data sent to activity modelling on remote server.

Although the system architecture has several advantages, the architecture is for the deployment stage in real situation while the focus of this thesis is the activity modelling and recognition component. That is the reason why the activity data collecting process in this experiment stores the activity data locally on Raspberry Pi instead of send them to a remote server.

The content of this chapter describes the methods and process for sitting activity modelling and recognition including:

- An overview of Hidden Markov Model (HMM), the modelling method used for activities (see section 5.1).
- The activity data collection, selection and pre-processing in order to build training dataset for HMM modelling (see section 5.2).

- The Model estimation. How to use the training dataset to retrieve the optimized parameter set of HMM (see section 5.3).
- The Model Evaluation. Evaluate the performance of the activity recognition component (see section 5.4).

5.1 Method for Hidden Markov Model Implementation

Hidden Markov Model (HMM) is a statistical modelling technique for modelling based on the assumption of Markov process (the prediction of future state is solely based on the present state, without knowing the full previous history). It is a popular modelling method for sequential data. “The sequential data is a kind of dataset that often arises through measurement of time series or arises in contexts, for example, the daily values of a currency exchange rate, or the sequence of nucleotide base pairs along a strand of DNA” (Bishop 2006, pp. 605).

In this thesis, the classified posture information based data that constructed in time-order is considered as sequential data, and it is assumed that one sitting activity is associated with a sequence of sitting postures. Meanwhile, this sequential data may contain information that has its own interstate transition probabilities for a specific HMM (an activity state, in this thesis). Based on those assumptions, the Hidden Markov Model is utilized for activity modelling. The content of this section introduces the concept of Hidden Markov Model and the strategy to implement this modelling method for activity recognition.

5.1.1 Overview of Hidden Markov Model

The HMM is a further improvement of Markov Process concept with two layers of stochastic processes. One underlying stochastic process is a non-observable hidden state layer. On top of this hidden layer, there is another stochastic process that could produce or emit the sequence of observed symbols by the hidden state layer (Rabiner and Juang 1986).

The relevant notation for includes:

- N represents the number of states in the model, individual state is denoted as $S_i = \{S_1, S_2, ...S_N\}$. It determines the size of matrix A which is $N \times N$.
- q_t means the states at time t, where $q_t \in \{S_1, S_2, ...S_N\}$.
- M represents the number of observations. Combined with N, it determines the size of matrix B which is $N \times M$.
- O represents the observation sequence, $O = O_1O_2...O_T$, where O_t means the observation at timestamp t.
- T, means the length of the observation sequence (total number of timestamps).

According to Rabiner's tutorial paper (Rabiner 1989), The HMM λ can be represented as

$$\lambda = (A, B, \pi)$$

The elements in this equation are:

- The probability matrix A represents state transition of the hidden states layer. $A = \{a_{ij}\}$, and a_{ij} means the state transition probability from S_i to S_j.
- The probability matrix B represents the stochastic process that emits the observation symbols. $B = \{b_{ij}\}$, where the b_{ij} is the probability of one particular observation O_j at the state S_i. It can be convert into a probability distribution function $b_i(O)$ of an observation vector O for a state S_i.
- π_i is the prior probability, represents the probability that a particular state S_i is the starting state.

The relation between the hidden state sequence, timestamp and observation sequence is shown in Figure 5-2 and could be represented as: the three elements have the same overall length T; at timestamp *t*, hidden state q_t is presented based on the matrix A and the previous hidden state q_{t-1}, and q_t produces one observation symbol O_t based on the matrix B. The correspondence of timestamp, hidden state and observation kneads the model parameters together. Thus, creating a HMM that correlates the time-order observation sequence and probability computation.

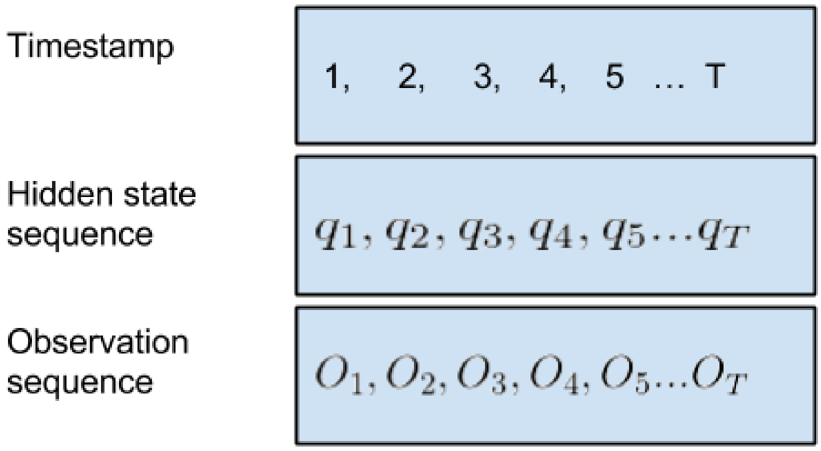


Figure 5-2 The relation of hidden state sequence, timestamp and observation sequence.

According to the description of HMM, the two most important parameters for setting up a HMM are N (number of hidden states) and M (number of observable symbols), because these two parameter determines the size of A and B while π is actually one of state set S. Since the HMM is constructed when N and M is assigned, how to determine the probability distribution inside the matrix of A and B through training or learning?

Before answering this question, Rabiner (Rabiner 1989) introduced three basic problems for HMM, they are:

1. Evaluation Problem: Given the observation sequence O and a model HMM λ, what is the probability of the $P(O|\lambda)$?
2. Decoding Problem: Given an observation sequence O and a model HMM λ, what is the best state sequence for this observation sequence?
3. Learning Problem: Given an observation sequence O, how to estimate the model parameters $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$?

Among these three problems, the Learning Problem (problem 3) answers the question for model training or learning, it utilizes an iterative Expectation-Maximization (EM) algorithm (Bilmes 1997), also known as the Baum-Welch algorithm to achieve a converged parameter set of λ, along with a given, observed training sequence.

In the training iteration process (current HMM compared with previous HMM) or after a set of HMMs (HMM for activity A compared with HMM for activity B) are trained, rises a question which is how to measure whether one model is better than another model based on a given sequence? The Evaluation Problem address this question and the probability $P(O|\lambda)$ of the observation set O and a given model λ is

$$P(O|\lambda) = \sum_{q_1 \dots q_T} \pi_{q1} b_{q1}(O_1) a_{q1q2} \dots a_{qt-1qT} b_{qT}(O_T)$$

This is called the likelihood of a model given a set of observations. Because it multiply many times value between 0 and 1(because two stochastic processes: state transition process based on A and observation emission process B are probability based), the value of $P(O|\lambda)$ could be potentially very small. So the result is usually represented by the logarithm value of $P(O|\lambda)$, which is called the log-likelihood (Rabiner 1989). In this thesis, the log-likelihood is a key metric for model selection and evaluation.

5.1.2 The Implementation of HMM with Activity modelling

Like other modelling process, there is a training stage before the model is used for prediction. In the training stage of a HMM for a specific activity, a modified posture sequence is input for learning process. This training posture sequence only contains the sequential posture information correlated to one particular activity. This training process is repeated to build up a set of independent HMMs where each HMM corresponds to a given activity (play games, relax, watch videos, etc.). Figure 5-3 shows the iterative training process of HMM. The final converged trained model will be saved for the prediction stage that follows.

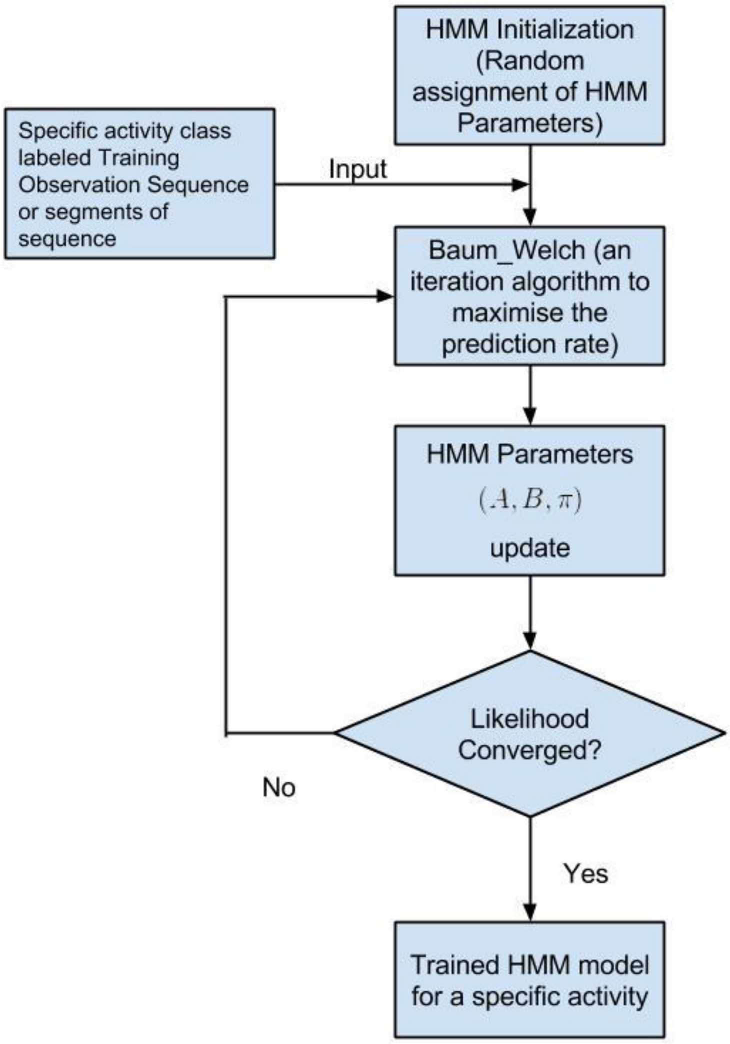


Figure 5-3 The training stage of HMM.

In the Figure 5-3, the input sequence data is the time ordered sequence of posture ID in this research, and it is used for both training and verification (the Baum Welch algorithm). In the initialization stage, random HMM parameters are assigned and the model utilise the Baum Welch algorithm to train itself and verify itself. When the iteration is over (the number of iteration can be set manually, it is 100 times in this experiment), if the prediction result (according to evaluation problem that is described in the last section) is higher than the beginning of the iteration, the HMM updates its three parameters, and then go over the training process again. If the prediction result converged (the result difference between the previous result and current result is lower than certain threshold), the whole training process stops, and the final converged trained model will be saved for the prediction stage that follows.

Before the prediction stage, it is assumed that a group of HMMs are trained and each HMM tackles one activity. In the prediction stage, each HMM in the trained model set takes a sequence of postures obtained from the previous classifiers as discrete inputs. Then, the system computes the set of probabilities for the input posture sequence across the set of HMMs generating an array of probabilities for corresponding activities. Finally, the input sequence is determined to be the activity model that has the highest log-likelihood in the array.

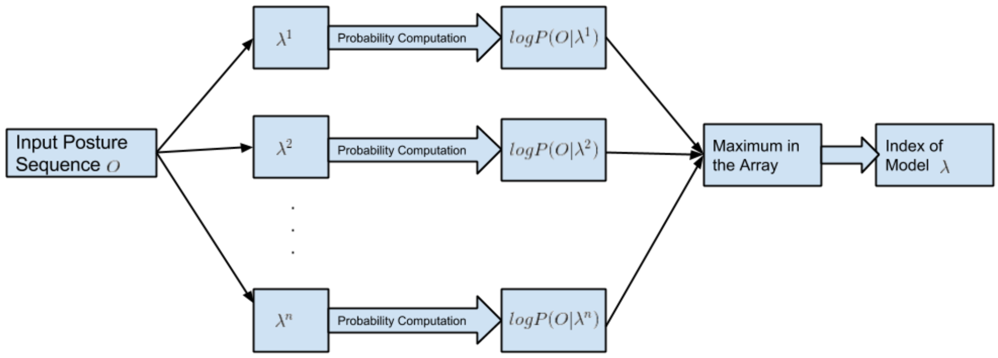


Figure 5-4 The Activity Recognition Process (Modified from (Rabiner 1989)).

Because the input is a time-ordered posture sequence, the time factor is also included in this modelling process. Through this modelling process, the posture sequence with certain length is abstracted into one activity symbol; thus, it compresses the whole posture sequence into an activity sequence. Hence, the relationship between posture and activity is established.

5.2 Activity Data Collection, Selection and Pre-processing

The aim of this section is to collect activity dataset, select meaningful activity in the dataset and furthermore convert the raw activity dataset into simplified posture information sequential data. The sequential dataset is divided into groups according to the activity. Each group of data is correlated to the training process for each activity Hidden Markov Model.

In this section, the content includes:

- Activity data collection: The experiment procedure to collect the activity data (see section 5.2.1).
- Activity data selection: Make choice on the activities based on consideration of coverage through subjects and amount of samples from each subject (see section 5.2.2).
- Activity data pre-processing: The conversion process that transforms the raw dataset from pressure sensor measurement into posture ID (different from pre-defined posture name) constructed sequential data for HMM training (see section 5.2.3).

Figure 5-5 shows the detail of data flow in the activity data collection, selection and pre-processing. The outcome of those processes is expected to be a group of posture sequences with a group of correlated activity label. Each sequence contains time-ordered postures which is the output of the posture classifier.

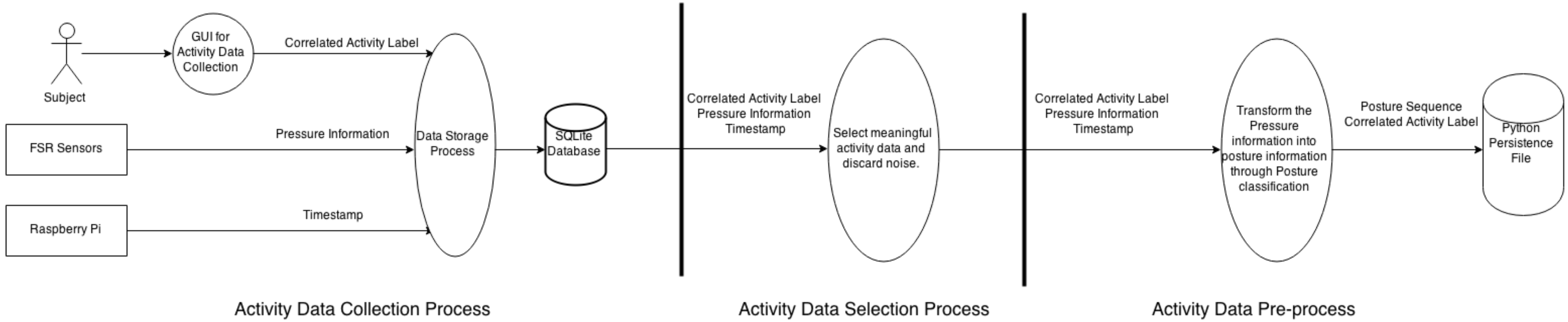


Figure 5-5 The data flow diagram for Activity data collection, selection and pre-process.
Activity Data Pre-process requires trained classifiers from section 4.4 to convert pressure information from FSR sensor to posture classes.

5.2.1 Activity Data Collection

The purpose of this data collection process is to collect a raw dataset for the sitting activity modelling. According to the discussion of HMM training, there are three elements that should be included in the activity training dataset:

- Timestamp for constructing the posture sequence in time-order.
- Data from FSR sensor for posture sequence construction.
- Correlated activity label.

The combination of those three elements at one timestamp is an activity data sample. The activity data samples will go through selection process (in section 5.2.2) and pre-processing process (in section 5.2.3). Additionally, according to the experiment design in section 3.3.5 and the discussion in section 3.2.5, the activity data collection should be as un-intrusive to participants' normal activity routine as possible and the participants should able to record their own activities.

To achieve those requirements, A Python based GUI program is developed to collect timestamp, pressure data from all 8 sensors and current activity. In this program, the participant's interaction with the GUI is all about activity. The participants can add new activities to the activity list, and select their current activity from the activity list. The pressure data and timestamp collection process is hidden in the back end in order to simplify the data collection process. To make sure the activity logged matches the real activity, a form named Activity Collection Form (see Appendix A) was given to the participant. If the participants made a mistake when using the GUI program (for example, clicked the wrong activity label while changing activity), they can write the mistake on this form with the correct activity label with its correct activity start time and end time. But none of those form is actually used by participants.

The reason that this program collects the pressure measurement from 8 sensors directly instead of using classified posture label is because the limitation of the Raspberry Pi's computation capability. The direct storage process allows the sampling frequency of the system to be maintained 6 Hz. This frequency allows the IntelliChair system able to collect all the posture changing information according to the experiment result (6.2 Hz) of experiment 4.3 with slightly difference. According to the development experience, if the classification process is involved between the data collecting and data storage process, the overall sampling frequency will decrease and can not reach 6 Hz frequency because the additional computation requirement for classification process. The raw pressure data conversion process will be performed in activity pre-processing section and the data will be transformed into posture information.

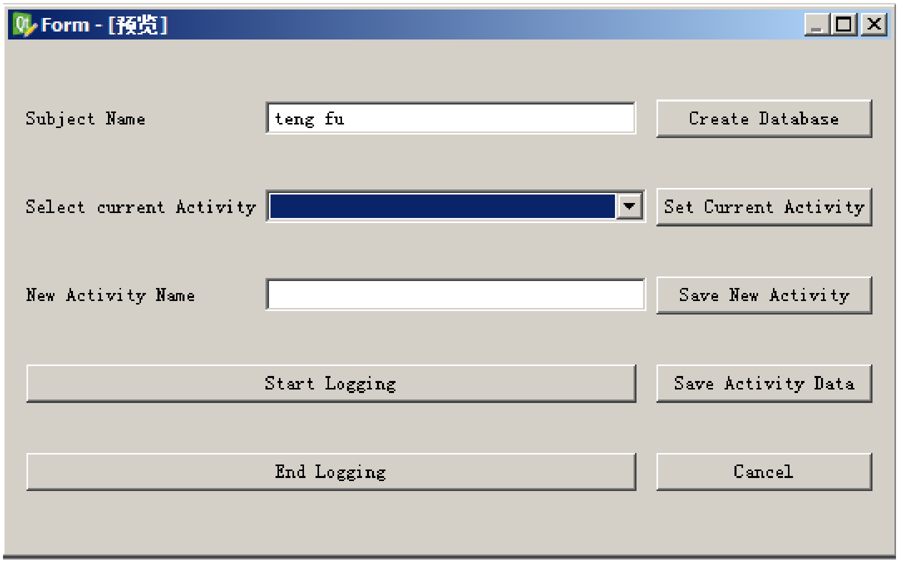


Figure 5-6 The GUI for activity data collection.

The experiment procedures are as follows:

1. Participants input their name in the Subject Name text field, and press Create Database button. This process creates a database that store all the related information.
2. Participants select the current activity from the activity list.
3. Participants click on the Start Logging button to start the data collection process.
4. The whole collection will last 4 hours. Within the 4 hours, the participants perform their normal sitting activity routine (e.g. reading, watching video). If the participants are going to change their activity, they just need to click the activity list and select the activity they are going to perform. If the participants find there is no appropriate activity in the activity list, they can type in the new activity name and press the Save New Activity button, the system will add the new activity into the list for the participants to select. The system will record

the information into the pre-created database.

5. When collection process is done, the experimenter click the End Logging button, the system will stop recording and the experiment is end.
6. Steps 1 to 5 will repeat for each participant.

Instead of a Python persistence file, all the activity data is stored in a database. This is because the long term experiment process will generate large amount of data, and database storage is more robust compared with persistence file storage. In order to reduce the time of database query process, the SQLite (The SQLite Development Team 2014) database, a light-weight, self-contained and transactional SQL database engine is selected for data storage.

The reason that this experiment would last 4 hours is because the experiment aims to collect naturally occurring activity data. In the real situation, subject usually performs few activities for one hour around and change to another activity. Additionally, if the experiment takes too much time, the number of participants will be low because they are not willing to take part in this experiment. In order to balance the number of participants and collect most of activity data, 4 hours of experiment time is selected because it is not too long for attracting the participants to get involved and within in this time interval, the participants are likely to perform several different activities.

The activity training dataset is ready after this experiment. There are total 10 participants (same subjects as in previous sitting posture experiment) in this experiment, so the total activity recorded is 40 hours. But not all the data is used in the experiment, and the reason is described in next section.

5.2.2 Activity Dataset Selection

The activity data collected in the last section aim to collect the naturally occurring postures and correlated natural activities. In order to pinpoint the correlation between the sitting posture and activity, the GUI program offers the ability that the subject could select pre-defined activities from the list or create their own activity if necessary. This design allows the subject to describe and record accurately their activity at a specific timestamp. But this can cause some problems that in the experiment period. Because different subjects may perform different activities, some of the activities can even be unique to one subject.

In order to discovery the activities with most coverage for different subject and the sufficient training samples, the filtering rules for activities selection are listed as follow:

(a) Select only activities that at least five subjects have performed.

(b) In the experiment, it is possible that subject left chair and go somewhere else (e.g. go to washing room). When this situation happens, the record in the database for all the sensor measurement is zero which is no use for activity analysis those record will be discard. This is the reason that the "vacant chair" activity is pre-defined in order to identify this situation that the subject has left chair which is easier for the data cleaning.

(c) The activity data with sample numbers that are less than 360 (it means 1 minute according to 6 Hz system sampling frequency) are discarded because it might be caused by trying the interface or miss-click the correct activity (e.g. working with PC activity of subject 7, watching video activity of subject 8.). This mainly because sometimes at the beginning of the experiment, subjects would click the activities in the drop list of the GUI just to experience the user interface, or sometimes, they click the wrong activity in the list and then realized in short time and then change back to the correct activity. But at the meantime, the system still records the activity label information. For data pre-processing purpose, those records are discarded.

(d) For the activity data with sample numbers larger than 5500, the samples in range of 100 to 5500 (total 5400) at the beginning is utilised. The reason is explained below.

According to the Hermann's research (Hermann 2005), a closure of capillaries occurs after around 15 minutes, which cause the body behaviour change for fatigued body relief; furthermore, it leads to a micro-motion change for the subject to maintain sitting posture. So the interval of activity data selection is chosen from 16 seconds after the activity started (sitting posture is stabilized) until 15 minutes later at 15 minutes 16 second (sitting posture is stabilized again). The sampling rate for activity data is 6 Hz, so the sample numbers of 15 minutes is 5400 while 16 second is 100. So the interval for sample selection is between 100 (16 seconds after activity starts) and 5500 (15 minutes and 16 seconds after activity starts). This interval includes the sitting postures when body is not fatigued and the beginning (16 second) of a micro-motion changing cycle for posture adjustment according to Hermann's research (Hermann 2005). With both the first time stabilized posture and re-stabilized posture information included; the information within this interval has better representation of the relation between activity and posture sequence information.

Activity	**Subject 1**	**Subject 2**	**Subject 3**	**Subject 4**	**Subject 5**	**Subject 6**	**Subject 7**	**Subject 8**	**Subject 9**	**Subject 10**
relax	29660	1459	472		6171	6207		6350		1691
vacant_chair	5783	21355	18635	10890	7451	2457	14654	3819		5048
reading	13700				39057	16		18		9771
working_with_PC'	6037	35242	22686	6155	36724	40381	15	46732	5877	34756
doodling				4695						
watching_video			429	64448		36791		20	10154	16459
play_games	19065	28160	49729				77185	6290	58986	9507
eating	11930	2549	959							4501
eating_and_watching_a_video			4470							
writing						321				
snooze										8022
Total/ Subject	86206	88812	97416	86210	89416	86193	91901	*93559*	75055	89770

(a)

Activity ID	**Activity**	**Subject 1**	**Subject 2**	**Subject 3**	**Subject 4**	**Subject 5**	**Subject 6**	**Subject 7**	**Subject 8**	**Subject 9**	**Subject 10**
1	**relax**	5400	1459	472		5400	5400		5400		1691
2	**working_with_PC'**	5400	5400	5400	5400	5400	5400		5400	5400	5400
3	**watching_video**			429	5400		5400			5400	5400
4	**play_games**	5400	5400	5400				5400	5400	5400	5400

(b)

Table 5-1 Table of activity data selection.
The numbers in the table are sample numbers. The definition of sample is a combination of timestamp, sensor data and correlated activity label.
(a) The total samples in activity and subject category.
(b) The selected samples in activity and subject category.

The activity data selection process in this thesis is specific for this experiment according to the hardware feature of IntelliChair. The final selection of the activity is listed in table 5-1 (b). They are Relax (1), Working with PC (2), Watching Video (3) and Play Games (4). Total four HMM will be trained and the relation between the HMM and activity are one to one, which means one HMM for one activity.

5.2.3 Activity Dataset Pre-processing

Because the training dataset for HMM estimation is the output of the posture classifiers that were trained in section 4.4, the content in this section is to pre-process the activity data and transform the data into a format that can be utilized for HMM training. The aim of this process includes:

- Convert the 8 channels of pressure measurement at one timestamp into one single integer represented posture using the classifier trained in section 4.4.
- Organize the posture sequence in time-order based on the correlated timestamp.
- Build an activity ID based label set that correlates with each posture sequence for further usage.

The procedure for this pre-processing is:

(a) Scale the raw FSR sensor measurement of all 8 channels.

(b) Split channel 1 to 4 data from the dataset as an input for the trained spine posture classifier.

(c) Split channel 5 to 8 data from the dataset as an input for the trained leg posture classifier.

(d) Transform the posture combination set $[P^s_{spine}, P^l_{leg}]$ into posture ID set $[P^i]$, and organize them in time order.

(e) Transform the activity names into activity ID as showed in Table 5-2 and organize them in time order.

Posture ID Transformation		**Leg Posture**											
		5	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Spine Posture	**0**	1	2	3	4	5	6	7	8	9	10	11	12
	1	13	14	15	16	17	18	19	20	21	22	23	24
	2	25	26	27	28	29	30	31	32	33	34	35	36
	3	37	38	39	40	41	42	43	44	45	46	47	48
	4	49	50	51	52	53	54	55	56	57	58	59	60

Table 5-2 The transformation between body part postures and the posture ID.

The reason for the process (d) is to simplify the posture input for HMM. Because there are two classifiers for sitting posture classification, so the result of classification consists of two posture numbers and the description for whole body posture is:

$$[P^s_{spine}, P^l_{leg}],\ s \in \{s \in \mathbb{N} : 1 \leqslant s \leqslant 4\},\ l \in \{l \in \mathbb{N} : 5 \leqslant l \leqslant 16\}$$

The whole body posture combination set is transformed into posture ID which is shown in Table 5-2. Because the four sensors on the back surface sometimes does not have contact with the subject which causes zero measurement, an additional fifth spine posture 0 is added in order to address the no contact on back issue. Since there are overall 60 possible spine and leg posture combinations, it means the number of observation symbols for HMM is 60.

The pre-process (e) is to build up a label set that is correlated with the HMM training set. The label set is not a part of HMM model estimation stage, but it is utilized in the HMM evaluation for the purpose of measure the accuracy of the prediction.

After the pre-process on the activity data, there are four groups of data. Each group correlates with a specific activity, and each group contains two datasets. One dataset is a time-order posture ID sequence, which is used to train a specific HMM for one activity. Another dataset is the label set, which is used for HMM evaluation.

5.3 Model Parameters Selection

In this section, the focus is the parameter estimation for the HMMs. Refer to the activity data selection result in section 5.2.2, there are total of four HMMs that will be trained which correlates with the four activities: Relax (1), Working with PC (2), Watching Video (3) and Play Games (4). In this thesis, the Hidden Markov Model code is a part of scikit-learn Python machine learning package (Pedregosa et al. 2011) which handles the model training and log-likelihood computation. The other related code is programed by thesis author.

Through the short overview of HMM structure in section 5.1.1, the HMM for each activity relies on the set of posture ID sequences, because the initial HMM parameter for each activity to recognize the posture sequence is unknown. The parameter set includes the specification of A, B , π , N and a new parameter which is the Length of the input observation sequence L. The new parameter L is used specifically for log-likelihood assessment, and the reason to introduce this parameter is explained below.

According to Rabiner’s tutorial (Rabiner 1989) and the experiments of Monekosso and Remagnino’s research (Monekossoand Remagnino 2009), the elements that are required to be determined for a HMM includes:

- The type of HMM. In this thesis, the type of model is assumed to be ergodic which means that every “hidden states of the model could be reached from every other state in finite number of steps” (Rabiner and Juang 1986).

- The hidden states number N. This parameter determines the size of matrix A and B.
- The input observation sequence length L. This is also called frame length in Rabiner's tutorial (Rabiner 1989). Because after the model is trained, the input for the model is a sequence of symbols, and this parameter is used to find out the best length of an input sequence in order to achieve the best prediction accuracy.

The rest of the parameters (A, B, π), will be automatically estimated and converged after the training process (Learning problem in section 5.1.1).

The parameter N is the first to be determined and the procedure for one HMM is:

(a) Use the K-fold cross-validation to split the whole training set (the posture ID sequence which is modified in section 5.2.3) into separate training and test set. The test set size is 10% of the whole training set.

(b) Use the separated training set to train a HMM with a fixed number of N and get the converged (A, B, π).

(c) Use the test set to feed the model and get the log-likelihood, which is a metric for a HMM with a specific test sequence.

(d) In this experiment, K = 10, which means the training and validating process repeat 10 times for a particular number of N.

(e) The average of log-likelihood of 10 times training and validating is the indicator of a HMM's performance under a particular number of N.

The range of N covers from 3 to 20 and this range is decided according to the research from Monekosso and Remagnino (Monekossoand Remagnino 2009). Within this range, the N that has the highest log-likelihood is selected. According to the experiment, the best N for each activity is listed in Table 5-3 and the log-likelihood distribution is shown in Figure 5-7.

Activity	**Best value of N**
Relax	N = 19
Working with PC	N = 20
Watching Video	N = 17
Play Games	N = 17

Table 5-3 The best   for each activity HMM.

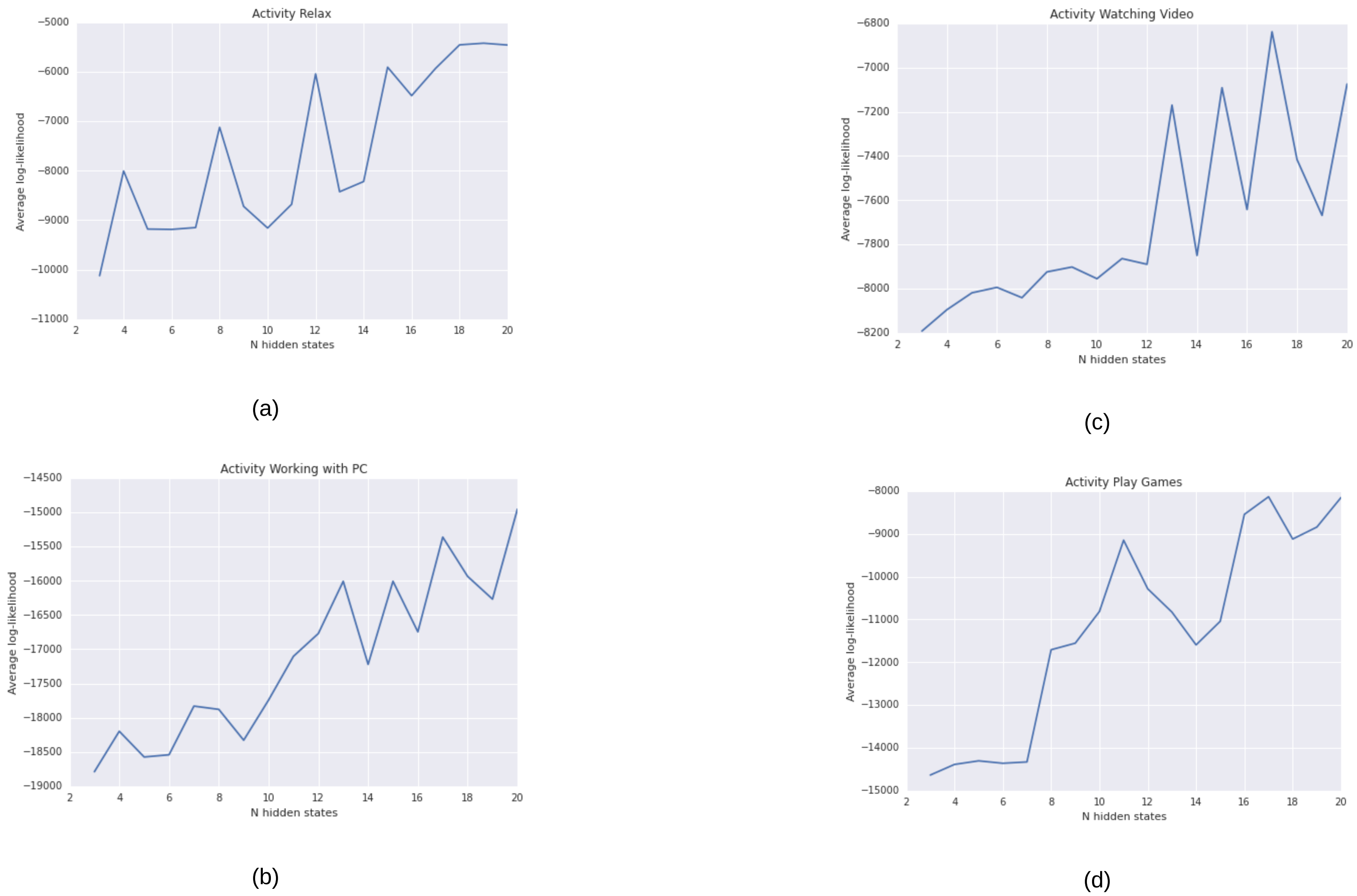


Figure 5-7 The average log-likelihood for different values of hidden state.(a) Relax (b) Working with PC (c) Watching Video (d) Play Games.

The next step is to find the best number of L based on the selected number of N. The procedure to perform the experiment is similar with the parameter N determining process, but the difference is step (c).

Instead of using the whole test set to feed the trained HMM, the fragment of test set is used to feed the trained HMM and get the log-likelihood for each fragment, and then the average of the log-likelihood for one particular L is retrieved as the metric for L selection. For example, assuming L is 6, then the whole test set is split into multiple fragments, and each of the fragments has the standard length of 6. Every fragment will be fed into the trained HMM and has an array of log-likelihoods. The average of the array is the performance indicator that whether this length of L is better or worse than a different number of L. The range of L is from $[6, 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102]$, which is the x-axis in Figure 5-8. The numbers in the range are all multiples of 6 because the sampling frequency of the IntelliChair system is 6 Hz. In the real situation, the best L length means in the prediction stage, how many seconds of data is adequate for feeding into the HMM for prediction. Figure.5-8 shows the L selection result.

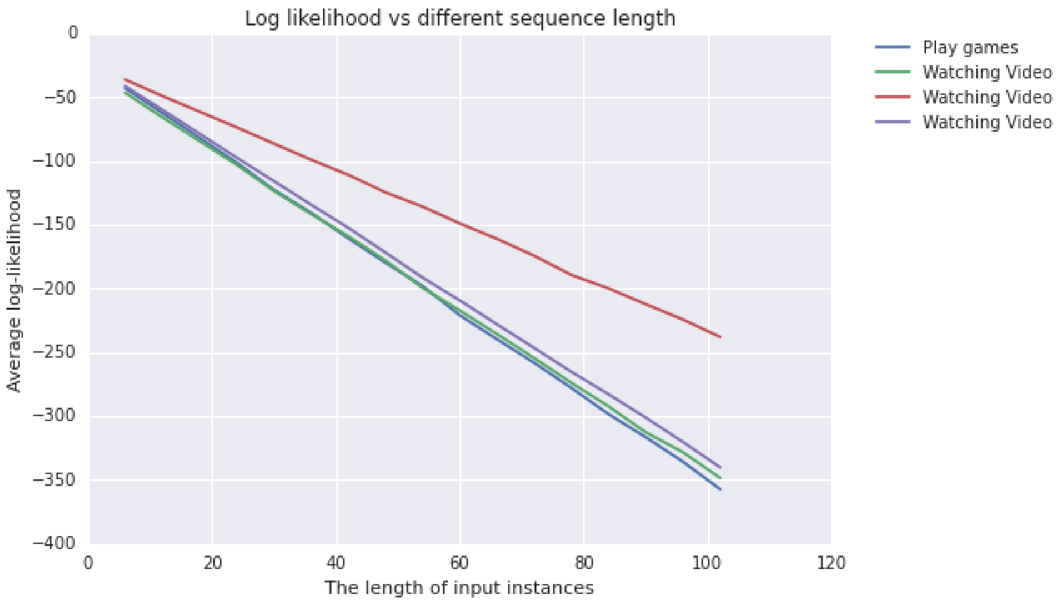


Figure 5-8 The log-likelihood vs different input sequence length. Length of 6 has the highest log-likelihood across four activities.

In this experiment, the best result for L is 6 across all four activities according to figure 5-8. It means the trained HMM could predict IntelliChair user's activity every one second, based on the incoming posture ID sequence. Refer to the experiment result of the N (Table 5-3) and L selection (Figure 5-8.), the best parameter set for activity modelling in this thesis is:

- Relax: N = 19, L = 6.
- Working with PC: N = 20, L = 6.
- Watching Video: N = 17, L = 6.
- Play Games: N = 17, L = 6.

In this section, a large amount of HMMs are trained, and among those HMMs, the HMMs with the highest log-likelihood for L in each activity are saved and they are constructed into an array for the activity recognition evaluation process in section 5.4.

5.4 Evaluation and Results

Since the HMMs for four activities are trained with optimized parameter set, the system is evaluated in this section. The evaluation strategy is as follows:

(a) Split the whole training data into a number of fragments, and all fragments with equal length of 6. Each fragment has a correlated true activity ID in the label set which was created in section 5.2.3.

(b) Feed every fragment into the model array which is created in section 5.3 and returns the predicted correlated activity ID. This is the prediction result for this model array to recognize the incoming fragment.

After the evaluation process, the confusion matrix is generated for the model array and it represents the activity recognition performance which is shown in table 5-4.

According to the evaluation result in table 5-4 (a) the system has an overall 41.27% accuracy for classifying the posture sequence into correct activity classes, for each activity:

- Relax: The system could recognize the posture sequence corresponding to this class with an accuracy of 51.83%.
- Working with PC: The system could recognize the posture sequence corresponding to this class with an accuracy of 50.33%.
- Watching Video: For posture sequence belonging to this class, the accuracy is 23.94%.
- Play Games: For posture sequence belonging to this class, the accuracy is 38.97%.

	Confusion Matrix for Activity Recognition					
	Data Set	**Relax**	**Working with PC**	**Watching Video**	**Play Games**	**Accuracy (%)**
True	**Relax**	**2178**	1303	66	656	51.820
	Working with PC	2290	**4077**	1087	646	50.333
	Watching Video	719	1303	**879**	770	23.944
	Play Games	2011	1424	410	**2455**	38.968
					Total	**41.266**

(a)

	precision	recall	f1-score	support
Relax	0.30	0.52	0.38	4203
Working with PC	0.50	0.50	0.50	8100
Watching Video	0.36	0.24	0.29	3671
Play Games	0.54	0.39	0.45	6300
Avg / total	0.45	0.43	0.43	22274

(b)

Table 5-4 The evaluation result of activity recognition.
(a) The confusion matrix of the activity recognition.
(b) The recognition performance for each activity.

The evaluation result shows the recognition accuracy is low and the reason causing the low accuracy is probably linked to two points: the HMM parameter setting and the activity data selection.

The HMM parameter setting for activity modelling is listed in the section 5.1. Among all the parameters, the number of observation symbol M is the likely factor for low accuracy. In this thesis, the number of M is defined to be 60 which is corresponding with the posture ID transformation process in the section 5.2.3. This transformation simplified the input posture sequence, but causes a high value of M.

The high value of M enlarge the size of probability matrix B which determines the stochastic process of observation production. The large size of B dilutes the observation emission probability, hence, it causes the HMM to be more susceptible to variation of input posture sequence. The Figure 5-9 shows the histogram of the posture ID distribution in each activity. As Figure 5-9 indicates, within 60 possible posture IDs, the frequency of some posture IDs might be low.

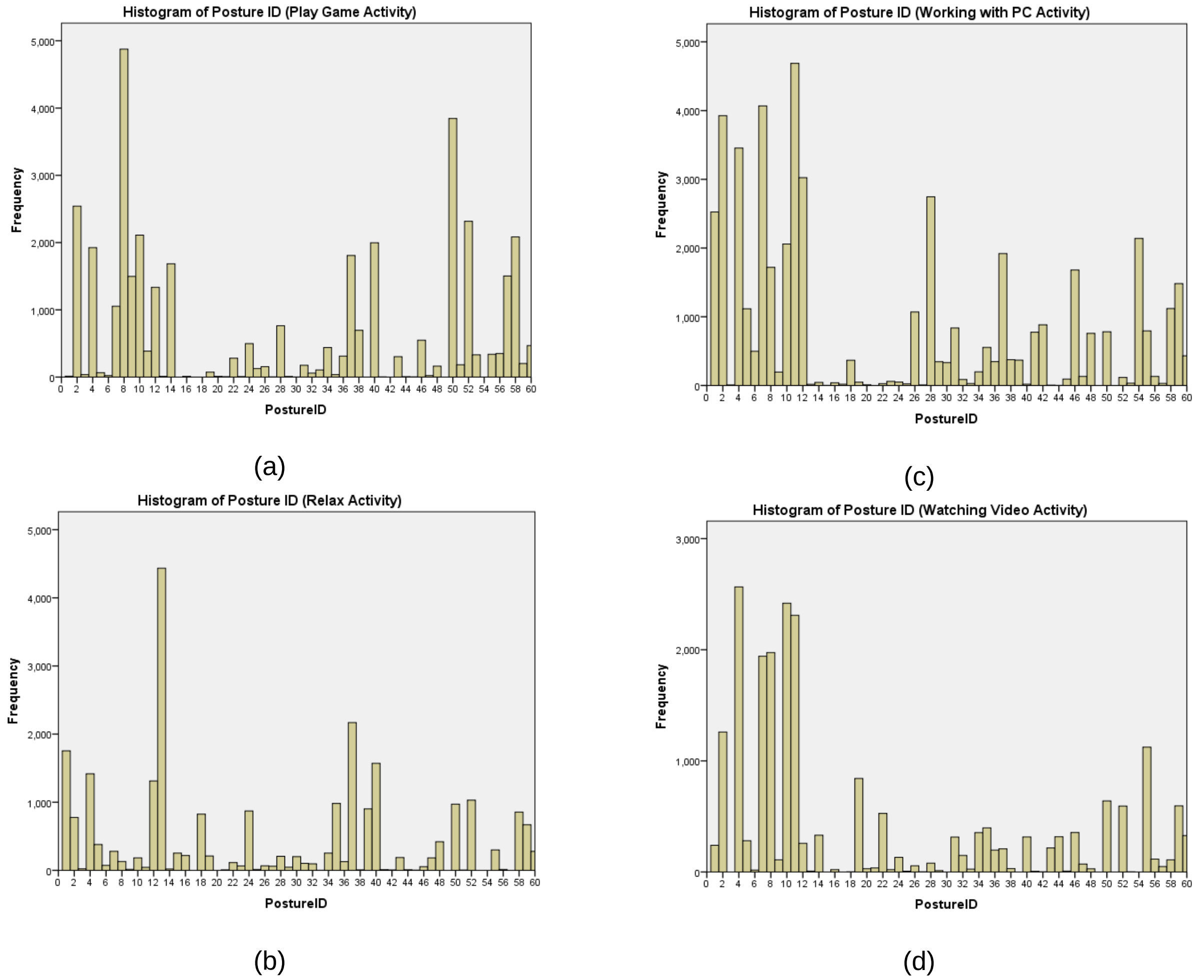


Figure 5-9 Histogram of Different Activity.
(a) Play Game (b) Relax (c) Working with PC (d) Watching Video.

Another possible reason for the low accuracy is the activity selection process. One thing needs to be emphasised is that in this experiment, all activity data is recorded by obtaining from natural activity routine and from multiple subjects. It is possible that the collected activity data contains a considerable level of noise. According to the confusion matrix in table 5-4 (a):

- Relax: Most confused sample comes from working with PC activity.
- Working with PC: Most confused samples are from Relax activity.
- Watching Video: Most confused samples are from working with PC activity, while confused samples from rest of the activities are also large.

- Play Games: Most confused samples are from Relax and working with PC activity.

Among the results, all four activities have significant overlap of confused samples set with each other. The explanation for the overlap is because of:

- The posture sequence for different activities from one subject is potentially similar. Take the Watching Video sample set as an example, its recognition accuracy is the lowest among those activities, which means it is possible that the posture sequence for in this activity is similar with Working with PC or Relax, and this feature of similarity is the same for different subjects. Hence, cause the overall recognition accuracy is not satisfactory.
- One subject's sitting habit for one activity might be similar to someone else's sitting habit of different activity. According to the observation of the experiment process, some subjects prefer casual sitting style (leaning back) when play games while some subject prefer an intense and focus sitting style (sitting forward or sitting on edge). The sitting posture differences might be influenced by the level of intense that the player engaged with PC. The differences potentially interfere with the recognition result, which can be seen from the confused sample set of Play Games activity. In this activity sample set, its major overlap samples is with Relax activity which means that some subjects in the training dataset used to sit casually to play games as well as relax, while others is used to focus and concentrate as well as working with PC.

- The incorrect and overlap annotation of activity. It happens in the experiment that the subject may have sometimes forget to interact with the GUI program to change the activity label when their actual activity changes. And sometimes the subject performed two activities at the same time (watching video and eating at the same time). Those actions are potentially another reason why the training data set for HMM may have incorrect activity labels that lower the recognition performance.

5.5 Summary

This chapter explains the construction and the evaluation of activity recognition component for the IntelliChair system. The activity recognition component analyse the time-ordered posture sequence which is generated by the posture classifiers, and use, the posture sequence to make prediction of current user's sitting activity.

For the construction part, the natural sitting activity data is collected along with the pressure data from all 8 sensors. Then it focuses on the HMM training for selected activities (Relax, Working with PC, Watching Video and Play Games) with most coverage across different subjects.

In order to train the HMM for each activity, the data is required to be pre-processed (the raw pressure data is classified into posture classes by utilizing the classifier that is trained in section 4.4) and transformed into posture ID (transform a two elements included array into one single integer) based sequence which simplifies the input data for HMM training. After all four HMMs are trained, the four models are combined together as an activity recognition component.

For evaluation part, the training dataset is split into fragments of equal length to evaluate the activity recognition component. The result is 41.27% for activity recognition accuracy. The potential reasons for the low accuracy may include the HMM parameter setting strategy (the number of observation postures), the different individual sitting habit, different subjects' sitting habit preferences when perform same activity (keep the same posture when performing different activities) and the potential incorrect and overlapping of activity labels in the experiment.

Chapter 6 Conclusions and Future Work

The content of chapter 6 includes the overview of this research in section 6.1, discussion and conclusion of the work in this research in section 6.2. Furthermore, future work of this research is discussed in section 6.3

6.1 Overview of Context Background

Section 6.1 describes the IntelliChair system that is built for this research and emphasizes the aim and the contribution of this research as well as the relevant experimental work.

6.1.1 Overview of System

IntelliChair, a non-intrusive, low hardware cost pressure sensing system is developed in this research order to investigate the sitting activity and the relationship between sitting postures and sitting activity. The IntelliChair system has eight sensors that are placed on the chair surface (four on the vertical surface and four on the horizontal surface) and the sensors are connected to a micro PC Raspberry Pi.

With micro PC Raspberry Pi integrated, the system is able to accomplish the posture classification task locally and the posture sequence information could be delivered to a remote server to fulfil the activity modelling and recognition task which requires more computational power. Figure 6-1 shows the separation of the two processes.

One thing to emphasize is that this is the potential deployment stage system architecture, and within the activity data modelling and recognition experiment, the collected activity data is firstly stored locally on Raspberry Pi, and then analysed on another desktop PC. The detail about the experiment procedure is described in section 5.2.

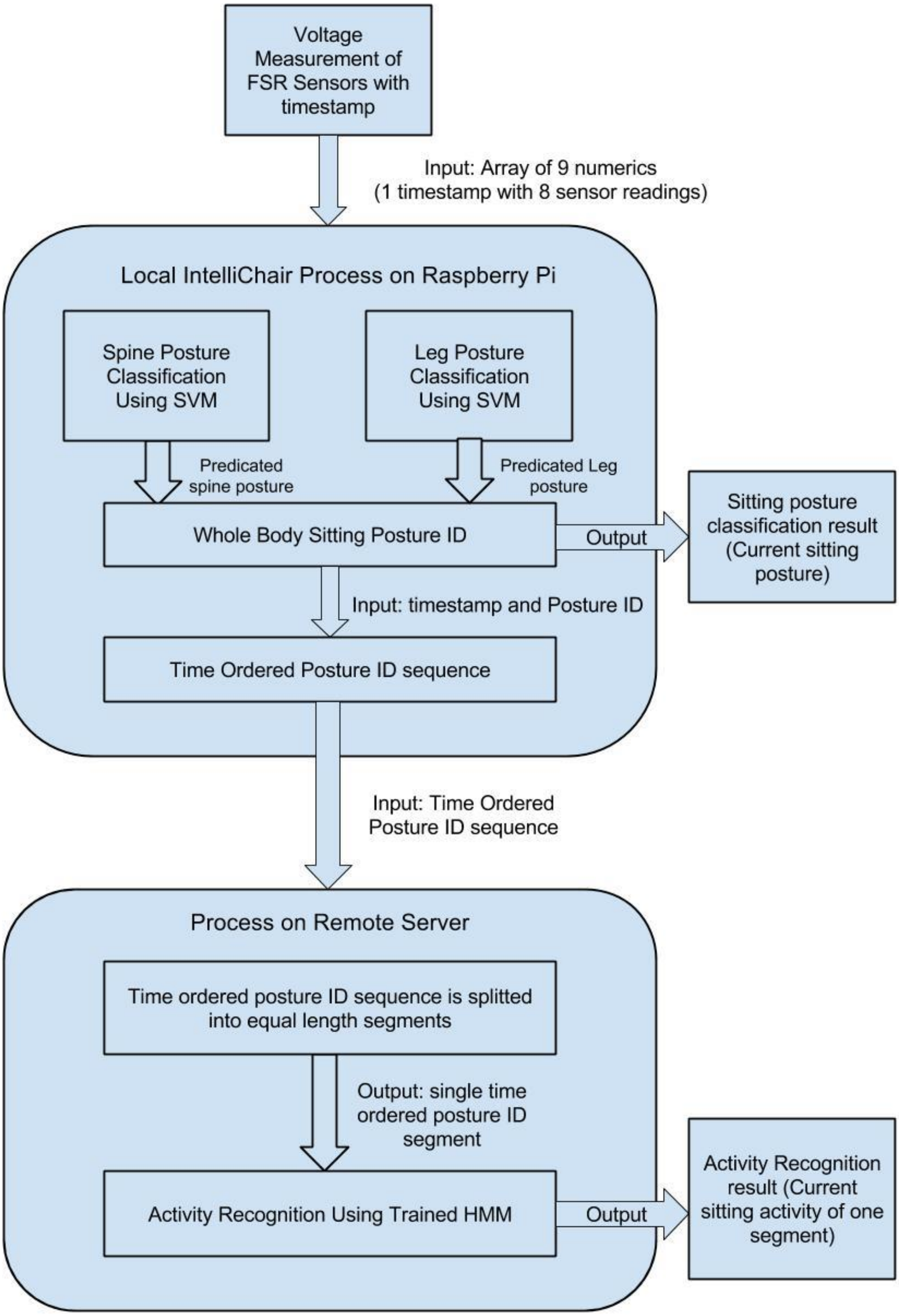


Figure 6-1 Posture classification on Local Raspberry Pi and Activity Recognition Process on Remote Server with detailed input and output data format.

With the two layers (one layer for posture classification locally, one layer for activity recognition remotely) system architecture, the IntelliChair can perform the posture monitoring task locally but also capable of integrating with Home level AmI or IE system such as CASAS (Center for Advanced Studies in Adaptive Systems). Compared with the low resolution of activity detection (CASAS system can detect which room the subject is located in the house and predict the subject's ADL such as cooking in kitchen), IntelliChair is able to provide high resolution activity data. For example, a subject is in the living room and is working with PC, in this situation, CASAS system is only able to detect the subject is in the living room. But with the assistance from IntelliChair, the CASAS system not only knows more detailed activity (output of the IntelliChair activity recognition system) about the subject, but also able to co-operate with subject by associating with subject's behaviour pattern.

6.1.2 Aim and Main Contribution

As introduced in section 1.1, the aim of this thesis is to build up a system that is capable of recognize sitting activity through a non-intrusive and low-cost sensor system. According to the description of the aim, the research objectives in this thesis are:

1. Develop a non-intrusive hardware system that is capable of collecting sensor data at a relatively small financial cost with high accuracy.
2. Detect the sitting posture through the collected data.
3. Establish a correspondence between sitting posture and sitting activities.

The main contribution of this thesis is based on the three objectives which include:

- The correct pressure sensors (Force Sensing Resistor) that are deployed at critical position could detect the sitting posture without compromising the posture classification accuracy. The finding of suitable sampling frequency for pressure based sitting activity detection system.
- Utilization of machine learning algorithm (Support Vector Machine) in sitting posture classification and achieved high accuracy performance.
- The finding of a correspondence between sitting activity (Relax, Working with PC, Watching Video and Play Games) and time-ordered posture sequence.

Ten participants took part in the research and according to the result, the IntelliChair system is able to classify the sitting postures with accuracy of 99.8% on spine postures and 99.9% on leg postures based on the posture data from those 10 subjects. When dealing with “unfamiliar” subject data, the accuracy are 80.7% for spine posture classification and 42.3% for leg posture classification. Furthermore, the IntelliChair system shows the possibility to recognize selected and pre-modelled four sitting activities based on the time-ordered posture sequence although the recognition performance is not satisfactory (41.27% overall).

6.1.3 Overview of Experimental Work

Five research questions that are extended from the objectives, which comprise:

1. (Objective 1a): What type of pressure sensor should the system use?
2. (Objective 1b): Are there any alternative sensors to achieve non-intrusive detection other than pressure sensing?
3. (Objective 1c): What is the signal characteristic of the pressure sensing information?

4. (Objective 2): How to classify the data in order to detect the sitting posture?
5. (Objective 3): How to build up the correspondence between the users' sitting posture and the users' activity?

Five experiments are performed in this research to answer each of the research questions above. The first experiment (see section 4.1) determined the FSR sensor as the pressure sensor for IntelliChair system. The second experiment (see section 4.2) validates the usability of vision based non-intrusive Kinect sensor as an alternative for pressure sensor. The third experiment (see section 4.3) utilizes the Fourier Transform method to discover the frequency characteristic information of posture signal. The fourth experiment (see section 4.4) collects sitting posture training data and estimates SVM based classifiers to recognize spine and leg postures. The fifth experiment (see section 4.5) collects natural occurred activity data and use HMM to build the correspondence between sitting posture sequence and sitting activity.

6.2 Conclusions and Discussions

6.2.1 Discussion on Pressure Sensor Technology

According to the discussion about previous sitting posture detection systems (Slivovsky and Tan 2000, Tan, Slivovsky and Pentland 2001, Mota and Picard 2003, Hermann and Koiva 2008, Mutlu et al. 2007, Cheng et al. 2013), there is a tendency of shifting from the high hardware cost (the price worth several thousands of pounds) of pressure array or matrix into low cost (several pounds each) pressure sensor but with optimized sensor position placement.

In general, the sensor number decrease may lead to posture classification accuracy decrease, but in this thesis, the author believe that better sensor placement strategy is able to balance the hardware cost and the posture classification performance and it is proved by the experiment result. In this thesis, author uses eight Force Sensing Resistors, four sensors on the vertical surface (for spine posture detection) and four sensors on the horizontal surface (for leg posture detection) which is based on Hermann's design (Hermann and Koiva 2008). Furthermore, author proposed an innovation in separation of spine and leg postures.

In previous research (Tan, Slivovsky and Pentland 2001, Mutlu et al. 2007, Mota and Picard 2003, Cheng et al. 2013), the whole body posture is represented by one pre-defined posture (usually only focus on the leg posture) which blurs the torso posture. The author's point of view is that through the separation of classification on spine posture and leg posture, each posture classifiers of IntelliChair system deals with only four dimensional data in training and classification process. Hence, the system is able to describe the whole body sitting posture more precisely by combining both postures from human torso part and human leg part.

Another contribution of this thesis is the determining of the sampling frequency for pressure sensor based posture detection system. No papers from previous researchers address this issue. In this thesis, the author proposed an experiment for posture signal characteristic analysis that is correlated with this study. Through frequency analysis of the posture signal information, the result of this experiment is retrieved which indicates that if a pressure sensor based posture detection system wants to detect normal sitting posture changes, the system sampling frequency is 6.2 Hz. If the system wish to detect sitting posture changes with potential anomalous activities (i.e. stroke), the sampling frequency should be at least 11 Hz. This result is particularly useful for follow-up researchers if they wish to develop such system with special purposes.

6.2.2 Discussion on Sitting Posture Classification

In order to collect the data for the experiments of sitting posture classification and sitting activity modelling and recognition, there are 10 subjects were invited to attend the data collection procedure. There are two stages for the data collection, firstly, the subjects are asked to perform 16 specific postures (4 spine postures and 12 leg postures). Secondly, the subjects will be asked to sit on the IntelliChair system for four hours, and perform their normal sitting activity routine and interact with the GUI program whenever their activity changes. The first part of the data is used for posture classification study, while the second part is used for sitting activity study.

For the study of sitting posture classification, the Support Vector Machine is utilized for posture classification to classify two set of postures (spine classifier and leg classifier). There are many classification algorithms are utilised in the previous research, and some of them shows good performance in classification. According to the compact and remote processing feature of the new hardware system, the author decided to select a new classification algorithm. This algorithm should be memory efficient and flexible to deal with difference situation, which is reason why the Support Vector Machine is determined and SVM proved itself in the experiment.

This classification accuracy achieves 99.8% on spine postures and 99.9% on leg postures when tested with subject's postures is included in the training set, but when a subject's posture data is not included, the accuracy decreases when dealing with this "unfamiliar" subject. The average classification accuracy for "unfamiliar" subject is 80.7% for spine postures and 42.3% for leg postures.

There first result is significant considering the data is collected with only 8 pressure sensors, and there are 16 postures. It proves the success that when the classifier is trained, the correct sensor placement still able to achieve promising posture classification performance with limited sensor numbers. But second result shows the side effect that this sensor placement is dependent on training data, which makes the classifier very sensitive to posture data from “unfamiliar” subject. The reason for this sensitivity is due to the different sitting habits and body characteristics of subjects. This feature determines the usage of the sitting posture classification component of the IntelliChair. In the author’s point of view, a training stage is necessary for the IntelliChair user before using the system (to train this posture classification component) in real situation. Considering the IntelliChair is mostly used in private situation, the drawback is acceptable.

6.2.3 Discussion on Sitting Activity Recognition

For the study of sitting activity modelling and recognition, the author of this thesis assumes that the time-ordered dynamic change of subject’s posture has information that related to subject’s sitting activities. Based on this assumption, the Hidden Markov Model (HMM) is used for activity modelling. One HMM represents one activity, and the posture sequence which is the time-ordered output of the posture classification component is the training and evaluation dataset for the HMM. Together with the four selected activities (Relax, Working with PC, Watching Video and Play Games) models, the set of HMMs is constructed into the activity recognition component. The reason for choosing these four activities is because those activities have most coverage across 10 subjects and they all have enough samples for HMM training.

The activity data is from the second stage of data collection procedure, and within this stage, the activity is labelled by the subjects themselves. After certain pre-process on the activity data (the transformation from raw pressure data into posture

ID based posture sequence), the posture sequence fragment with length of 6 (one second posture sequence) is fed into this component and this posture sequence fragment will be classified into one of the four activity categories.

The activity recognition results were 51.82% for Relax, 50.33% for Working with PC, 23.94% for Watching Video and 38.97% for Play Games. With only 41.27% of overall activity recognition accuracy, the result is not satisfactory. The reasons for the low accuracy include:

- The substantial spine and leg posture combination affect the quality of HMM training.
- The different sitting habit from different subjects interfere each other.
- For one subject, the posture sequence for different activities might be similar.

Further work is needed to seek the improvement based on simplification of posture sequence (the decrease number of posture ID occurred) because some of the 60 possible back and leg posture combinations have very low rate of occurrence (see Figure 5-9 in section 5.4). Meanwhile, further clarification and separation between different activities and subject's data is another direction of improvement.

For conclusion, the work of this thesis satisfied its aim which is to build up a system that is capable of recognize sitting activity through detected sitting posture based on a non-intrusive and low-cost sensor system. First, the low cost and non-intrusive FSR sensor is selected for IntelliChair system and a reasonable sensor placement is chosen based on the trade-off between sensor numbers and posture classification result. Second, the IntelliChair system has a reliable posture classification performance when dealing with “familiar” subject after its posture classification component is trained. Third, for sitting activity recognition component, although only Relax and Working with PC two activities have a relatively better recognition performance, these two activities could be a symbol whether the IntelliChair user is in intense activity or taking a break. Thus, the IntelliChair system is still able to provide limited information that whether an IntelliChair user is in a state of leisure or engaging with computers.

6.3 Future Work

The IntelliChair system that is developed in this research has the potential to be a part of larger Intelligent Environment or Ambient Intelligence system. This leads to many questions for future work.

For further system improvement, there are three future work goals, including:

- The improvement on hardware design.
- The improvement on the posture classification generalization capability.
- The improvement on the activity recognition performance.

The hardware design improvement, in particular, is the integration of the IntelliChair hardware components, because current IntelliChair is still assembled on a wooden office chair. The goal of the integration could refer to Hermann’s tactile Chair

(Hermann and Koiva 2008). All the sensors, support circuit and the Raspberry Pi could be embedded into the protective mat, making the system a mat based system that has certain mobility, which is chair independent. This improvement would give the IntelliChair the potential to be flexibly deployed on any chair surface.

Another vision for IntelliChair is the improvement of the Raspberry Pi. Recently, a new Raspberry Pi model was released with upgraded processor (700 MHz to 900 MHz) and RAM (256 M to 1G). It is possible to upgrade the IntelliChair system with the new Raspberry Pi model in near future. The impact of the local micro PC replacement is the potential that the activity recognition process could be shifted from remote server to new Raspberry Pi. This change makes the IntelliChair is able to perform both sitting posture classification and activity recognition task locally, thus, making the IntelliChair more independent from the remote server, and it is possible for IntelliChair system to communicate with local AmI system (e.g. CASAS system in section 1.1) through a local network instead of internet access.

The objective of the improvement on the posture classification generalization capability is to make the classification performance better when IntelliChair is dealing with “unfamiliar” subject posture data. In particular, the improvement requires substantial posture data from multiple subjects with different body characteristics to enrich the training set for posture classification component.

The improvement on the activity recognition performance has two directions:

- The simplification of the posture sequence as HMM training data. Decreasing the number of posture symbols occurring (60 posture IDs in this thesis) in the posture sequence, could narrow the size of the observation emission matrix M in the HMM, hence, improving the recognition performance of the HMM.
- More pronounced definition or clarification between different activities. In this thesis, the activity labels are setup by the subject themselves, and the subjects sometimes might combine some activities together (in this thesis, some subjects combined eating and watching video together) which is potentially the noise source of natural activity data. Thus, a more clear definition between activities might lead to a more precise HMM training dataset for one activity that improves the HMM recognition performance. This is the main direction for performance improvement.

The improvement above is focusing on the IntelliChair system, beyond the IntelliChair system itself, it has the potential to be integrated in to current development of Ambient Intelligence or Intelligent Environment system such as CASAS in section 1.1 (Cook et al. 2013). Because IntelliChair recognizes the sitting activities while AmI or IE systems recognize daily living activities (cooking, eating, etc.), the level of activity description is different, where IntelliChair provides more activity information with higher resolution. With the higher resolution activity data, current AmI or IE systems is possible to extend their services with additional or better functionalities. Even more, the IntelliChair could make contributions to the arising of Internet of Things which communicates with other smart devices directly.

For instance, imaging there is a home monitoring system for elderly care which allows the care givers to monitor elderly in their home in order to reducing hospitalization cost through early intervention and treatment of stroke. Without a camera or body sensor network, the low resolution activity data from AmI or IE system is less likely to detect the stroke. Conversely, devices like camera or body sensor network are either privacy intrusive or body intrusive to elder, especially in the home environment. IntelliChair is possible to fit into the gap between the privacy and abnormal activity detection. Because IntelliChair is designed based on non-intrusive sensing so it does not need to be attached to human body, furthermore, by increasing its sampling frequency (higher than 11 Hz as discussed in section 4.3), it is possible for IntelliChair to recognize the abnormal pattern of stroke activity.

Apart from the home environment, IntelliChair can be potentially used by other indoor environment situation that deals with multiple subjects such as office. IntelliChair can be used for sitting posture and sitting activity monitoring and give feedback to the subjects in order to prevent fatigues or potential harm to human body in long term sitting. In that situation, the IntelliChair is possibly to deal with multiple subjects. Since the IntelliChair is designed as a plug and play smart device, and with the support from Radio Frequency Identification (RFID) technology and cloud computing, IntelliChair could utilise RFID to identify the current subject that is sitting on IntelliChair and would be able to fetch the subject's sitting posture classifier and sitting activity model from the cloud and perform the individual service. It is likely that the multiple subject or user situations is possible to cause privacy issues, since it is not the major focus of this thesis, so there is no further discussion on this topic.

6.4 Final Statements

This research presented the process of hardware development, data collection and data analysis of a sensor system named IntelliChair. The IntelliChair system can recognize the user's sitting postures; furthermore, infer the user's sitting activity based on the context of posture information. The IntelliChair was tested with a low-cost but reliable sensor placement framework, and it is also capable to integrate with other sensor system (i.e. Kinect sensor). Throughout this thesis, this research provides a detailed research procedure for sitting posture and sitting activity modelling and recognition.

Appendix A Activity Collection Form for activity adjustment

As described in section 5.2.1, this form is used by participant in the situation that they record wrong or inaccurate data in GUI of experiment.

Activity Collation Form

for subject_________

Start Time	End Time	Activity	Comment

Notice: This form is used when you find yourself miss-clicked the wrong activity label in the GUI, or the activity in the list could not describe your current activity, or you are performing multiple activities at the same time.

References

AB Electronics UK 2013. *ADC Pi - Raspberry Pi Analogue to Digital Converter.* [online]. Available from: https://www.abelectronics.co.uk/products/3/Raspberry-Pi/17/ADC-Pi-V2---Raspberry-Pi-Analogue-to-Digital-converter [Accessed 20 April 2013].

Anttonen, J. and Surakka, V., 2005. Emotions and heart rate while sitting on a chair: *Proceedings of the SIGCHI conference on Human factors in computing systems* 2005. Portland, April 02 -07. New York: ACM

Arduino 2012. *Arduino - Home.* [online]. Available from: http://www.arduino.cc/ [Accessed 10 July 2012].

Arnrich, B. et al. 2010. What does your chair know about your stress level? *IEEE Transactions On Information Technology in Biomedicine.* 14(2): pp.207-214.

Augusto, J. C. et al. 2013. Intelligent environments: a manifesto. *Human-Centric Computing and Information Sciences.* 3(1): pp.1-18.

Bilmes, J.A., 1997. *A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models.*TR-97-021. International Computer Science Institute.[Online] Available from: ftp://ftp.icsi.berkeley.edu/pub/techreports/1997/tr-97-021.pdf [Accessed 15 May 2012].

Bishop, C. M. 2006. *Pattern recognition and machine learning.* New York: Springer.

Brady, S., et al., 2006. Combining wireless with wearable technology for the development of on-body networks: International Workshop on *the Wearable and Implantable Body Sensor Network*,2006. *BSN 2006.,* April 03-05 2006. London: IEEE, pp. 4-36

Brady, S., Diamond, D. and Lau, K. T. 2005. Inherently conducting polymer modified polyurethane smart foam for pressure sensing. *Sensors and Actuators A: Physical.* 119(2): pp.398-404.

Cheng, J., et al., 2013. Smart Chair: What Can Simple Pressure Sensors under the Chairs' Legs Tell Us about User Activity?: *UBICOMM 2013, The Seventh International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies* 2013. Porto, September 29 - October 3. Wilmington:IARIA XPS Press, pp. 81-84.

Choi, S. and Jiang, Z. 2006. A novel wearable sensor device with conductive fabric and PVDF film for monitoring cardiorespiratory signals. *Sensors and Actuators A: Physical.* 128(2): pp.317-326.

Ciampone, S. 2012. *Methods for seated posture recognition.* PhD Thesis. University of London.

Cook, D., et al., 2009. Collecting and disseminating smart home sensor data in the CASAS project: *Proceedings of the CHI Workshop on Developing Shared Home Behavior Datasets to Advance HCI and Ubiquitous Computing Research* 2009. Boston, April 04-09 New York: ACM

Cook, D. J., et al., 2013. CASAS: A smart home in a box. *Computer*. 46(7): pp.62-69.

Corazza, S. et al. 2006. A markerless motion capture system to study musculoskeletal biomechanics: Visual hull and simulated annealing approach. *Annals of Biomedical Engineering.* 34(6): pp.1019-1029.

Cortes, C. and Vapnik, V. 1995. Support-vector networks. *Machine Learning.* 20(3): pp.273-297.

Du, H. 2010. *Data mining techniques and applications: An introduction.* Andover: Course Technology Cengage Learning.

Dunne, L. E. et al. 2005. Initial development and testing of a novel foam-based pressure sensor for wearable sensing. *Journal of Neuroengineering and Rehabilitation.* 2(1): pp.4.

Forlizzi, J., et al., 2005. The SenseChair: The lounge chair as an intelligent assistive device for elders: *Proceedings of the 2005 conference on Designing for User eXperience,* November 03-05 2005. New York: AIGA, pp. 1- 13.

Fu, T. and Macleod, A., 2014. IntelliChair: An Approach for Activity Detection and Prediction via Posture Analysis: *Proceeding of Intelligent Environments (IE), 2014 International Conference on* 2014. Shanghai, June 30-July 04, 2014. London: IEEE, pp. 211-213.

Gunn, S.R., 1998. *Support vector machines for classification and regression.* University of Southampton, School of Electronics and Computer Science.[Online] Available from: http://users.ecs.soton.ac.uk/srg/publications/pdf/SVM.pdf [Accessed 20 May 2014].

Heo, J., Kim, K. and Lee, J. 2009. Development of a Distributed Force Detectable Artificial Skin Using Microbending Optical Fiber Sensors. *Journal of Intelligent Material Systems and Structures.* 20(17): pp.2029-2036.

Hermann, S. 2005. Exploring sitting posture and discomfort using nonlinear analysis methods. *IEEE Transactions On Information Technology in Biomedicine.* 9(3): pp.392-401.

Hermann, T. and Koiva, R. 2008. tacTiles for ambient intelligence and interactive sonification. In: Anon. *Haptic and audio interaction design.* New York: Springer. 2008, pp. 91-101.

Hermann, T. 2006. *Thomas Hermann::Publication::Media Files.* [online]. Available from: http://www.sonification.de/publications/HermannKoiva2008-TFA/ [Accessed 20 June 2012].

Hsu, C., Chang, C. and Lin, C. 2010. *A practical guide to support vector classification*. [online]. Available from: https://www.cs.sfu.ca/people/Faculty/teaching/726/spring11/svmguide.pdf [Accessed 10 July 2013].

Hsu, C. and Lin, C. 2002. A comparison of methods for multiclass support vector machines*. IEEE Transactions On Neural Networks.* 13(2): pp.415-425.

Interlink Electronics 2006. *FSR Integration Guide & Evaluation Parts Catalog With Suggested Electrical Interfaces.* [online]. Available from: https://www.sparkfun.com/datasheets/Sensors/Pressure/fsrguide.pdf [Accessed 11 July 2012].

Junnila, S., Akhbardeh, A. and Värri, A. 2009. An electromechanical film sensor based wireless ballistocardiographic chair: Implementation and performance. *Journal of Signal Processing Systems.* 57(3): pp.305-320.

Latré, B. et al. 2011. A survey on wireless body area networks. *Wireless Networks.* 17(1): pp.1-18.

Lekkala, J. and Paajanen, M., 1999. EMFi-New electret material for sensors and actuators: *Proceedings of 10th International Symposium on Electrets* 1999. Athens, , September 22-24, 1999. London: IEEE, pp. 743-746.

Liechti, C. 2013. *Welcome to pySerial's documentation -- pySerial 2.6 documentation.* [online]. Available from: http://pyserial.sourceforge.net/ [Accessed 15 February 2013].

Lo, B. and Yang, G., 2005. Architecture for body sensor networks: *Perspectives in Pervasive Computing*, pp. 23-28.

Lo, B. et al. 2005. *Body sensor network - a wireless sensor platform for pervasive healthcare monitoring.* [online]. Available from: http://csis.pace.edu/~marchese/CS396x/L3/p077-080.pdf [Accessed 15 February 2011].

Lobo, J. M., Jiménez-Valverde, A. and Real, R. 2008. AUC: a misleading measure of the performance of predictive distribution models. *Global Ecology and Biogeography.* 17(2): pp.145-151.

Lueder, R. and Noro, K. 1994. *Hard facts about soft machines: The ergonomics of seating.* London: CRC Press.

Manunza, I. and Bonfiglio, A. 2007. Pressure sensing using a completely flexible organic transistor. *Biosensors and Bioelectronics.* 22(12): pp.2775-2779.

Marenzi, E., et al., 2012. Design and development of a novel capacitive sensor matrix for measuring pressure distribution: *Proceeding of Sensors Applications Symposium (SAS), 2012.* Brescia, February 07-09, 2012. London: IEEE, pp. 1-6.

Measurement Computing Corporation 2015. *USB DAQ Device with Eight 12-Bit Analog Inputs - Measurement Computing.* [online]. Available from: http://www.mccdaq.com/usb-data-acquisition/USB-1208LS.aspx [Accessed 10 October 2012].

Microsoft 2015. *Kinect for Windows.* [online]. Available from: https://www.microsoft.com/en-us/kinectforwindows/ [Accessed 20 February 2013].

Microsoft 2013. *PyKinect – write Kinect games using Python!* [online]. Available from: https://github.com/Microsoft/PTVS/wiki/PyKinect [Accessed 10 January 2013].

Moeslund, T. B., Hilton, A. and Krüger, V. 2006. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding.* 104(2): pp.90-126.

Monekosso, D. N. and Remagnino, P. 2009. Anomalous Behavior Detection: Supporting Independent Living. In: Monekosso, D. N., Remagnino, P. and Kuno, Y., eds. *Intelligent Environments: Methods, Algorithms and Applications*. London: Springer. 2009, pp. 35-50.

Mota, S. and Picard, R.W., 2003. Automated posture analysis for detecting learner's interest level: *Proceeding of Computer Vision and Pattern Recognition Workshop, 2003. CVPRW'03. Conference on* 2003. Madison, June 16-22, 2003. London: IEEE, pp. 49-49.

Mutlu, B., et al., 2007. Robust, low-cost, non-intrusive sensing and recognition of seated postures: *Proceedings of the 20th annual ACM symposium on User interface software and technology* 2007. Newport, October 07-10, 2007. New York: ACM, pp. 149-158.

National Instruments Corporation 2015. *NI LabVIEW - improving the productivity of Engineers and Scientists.* [online]. Available from: http://www.ni.com/labview/ [Accessed 10 September 2012].

National Science Foundation 2012. *VPython 3D Programming for Ordinary Mortals.* [online]. Available from: http://vpython.org/ [Accessed 25 October 2012].

Numpy Developers 2013. *Numpy - Numpy.* [online]. Available from: http://www.numpy.org/ [Accessed 2013].

Obdrzalek, S., et al., 2012. Accuracy and robustness of Kinect pose estimation in the context of coaching of elderly population: *Engineering in medicine and biology society (EMBC), 2012 annual international conference of the IEEE* 2012. San Diego, August 28-September 01, 2012. London: IEEE, pp. 1188-1193.

Pedregosa, F. et al. 2011. Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research.* 12: pp.2825-2830.

Putnam, W. and Knapp, R. B. 1996. Input/data acquisition system design for human computer interfacing. *Unpublished Lecture Notes, October.* 17.

Python Software Foundation 2015. *Welcome to Python.org.* [online]. Available from: https://www.python.org/ [Accessed 15 February 2012].

Rabiner, L. and Juang, B. 1986. An introduction to hidden Markov model., *IEEE ASSP Magazine.* 3(1): pp.4-16.

Rabiner, L. R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE.* 77(2): pp.257-286.

Raspberry Pi Foundation 2013. *Raspberry Pi.* [online]. Available from: https://www.raspberrypi.org/ [Accessed 10 March 2013].

Rothmaier, M., Luong, M. P. and Clemens, F. 2008. Textile pressure sensor made of flexible plastic optical fibers. *Sensors.* 8(7): pp.4318-4329.

Sarle, W. 27 March 2014. *Should I normalize/standardize/rescale the data?* [online]. Available from: http://www.faqs.org/faqs/ai-faq/neural-nets/part2/section-16.html [Accessed 20 October 2013].

Scholkopft, B. and Mullert, K., 1999. Fisher discriminant analysis with kernels: *Proceedings of the 1999 IEEE Signal Processing Society Workshop Neural Networks for Signal Processing IX,* Madison, August 23-25, 1999. London: IEEE, pp. 41-48.

Slivovsky, L. A.Tan, H. Z. 2000. *A real-time sitting posture tracking system.* PhD Thesis. Purdue University.

Smith, S. 2013. *Digital signal processing: A practical guide for engineers and scientists: A practical guide for engineers and scientists.* Edinburgh: Newnes.

Someya, T. et al. 2004. A large-area, flexible pressure sensor matrix with organic field-effect transistors for artificial skin applications. *Proceedings of the National Academy of Sciences of the United States of America.* 101(27): pp.9966-9970.

Stirling, D., et al., 2011. Objective functional capacity assessment using inertial sensorIn: *Healthcare Informatics, Imaging and Systems Biology (HISB), 2011 First IEEE International Conference on* 2011. San Joe, July 26-29, 2011. London: IEEE, pp. 272-277.

Tan, H. Z., Slivovsky, L. A. and Pentland, A. 2001. A sensing chair using pressure distribution sensors. *IEEE/ASME Transactions On Mechatroni,.* 6(3): pp.261-268.

Tan, H. Z. 1999. A sensing chair. *Proceedings of the ASME Dynamic Systems and Control Division.* 67: pp.313-317.

Tekscan 2015. *Body Pressure Measurement System (BPMS) - Research.* [online]. Available from: https://www.tekscan.com/products-solutions/systems/body-pressure-measurement-system-bpms-research [Accessed 15 November 2012].

Tessendorf, B., et al., 2009. Unsupervised monitoring of sitting behaviour: *Proceedings of Engineering in Medicine and Biology Society*, 2009. Minneapolis, September 02-06, 2009. London: IEEE. pp. 6197-6200.

The SQLite Development Team 2014. *SQLite Home Page.* [online]. Available from: https://www.sqlite.org/index.html [Accessed 10 April 2014].

Tscheligi. M., Markopoulos. P., Wichert. R. and Mirlacher. T. 2009. Ambient Intelligence: European Conference: *Proceedings of AmI 2009*. Salzburg, November 18-21, 2009. New York: Springer.

Wang, L. et al. 2012. A hierarchical approach to real-time activity recognition in body sensor networks. *Pervasive and Mobile Computing.* 8(1): pp.115-130.

Weiser, M. 1991. The computer for the 21st century. *Scientific American.* 265(3): pp.94-104.

Weisstein, E. W. 1999. *Sampling theorem*. [online]. Available from: http://mathworld.wolfram.com/SamplingTheorem.html [Accessed 10 February 2014].

Witten, I. H. and Frank, E. 2005. *Data mining: Practical machine learning tools and techniques.* San Francisco :Morgan Kaufmann.

Yang, S. and Li, Q. 2012. Inertial sensor-based methods in walking speed estimation: A systematic review. *Sensors.* 12(5): pp.6102-6116.

Zeng, H. and Zhao, Y. 2011. Sensing movement: Micro sensors for body motion measurement. *Sensors.* 11(1): pp.638-660.